

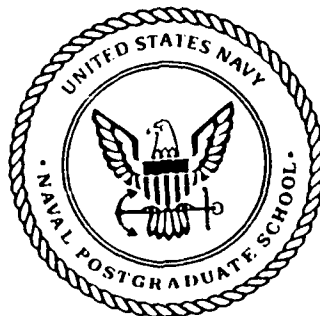
2

NPS-62-90-009

# NAVAL POSTGRADUATE SCHOOL

## Monterey, California

AD-A232 733



DTIC  
ELECTE  
MAR 12 1991  
S B D

### THESIS

AN ANALYSIS OF MLAYER: A MULTILAYER  
TROPOSPHERIC PROPAGATION PROGRAM

by

Lean-Weng Yeoh

June 1990

Thesis Advisor:

Hung-Mou Lee

Approved for public release; distribution is unlimited

Prepared for: The Chief of Naval Operations (OP-03B)  
Pentagon  
Washington, DC 20350

91 3 06 027

NAVAL POSTGRADUATE SCHOOL  
Monterey, California

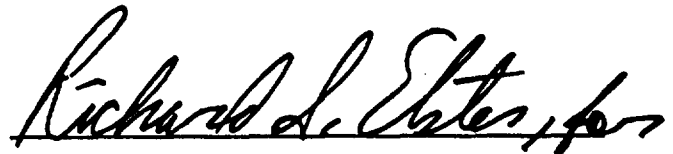
Rear Admiral Ralph W. West, JR.  
Superintendent

Harrison Shull  
Provost

This report was prepared in conjunction with research conducted for the Chief of Naval Operations (OP-3B), Pentagon, Washington, DC 20350 and funded by the Naval Postgraduate School.

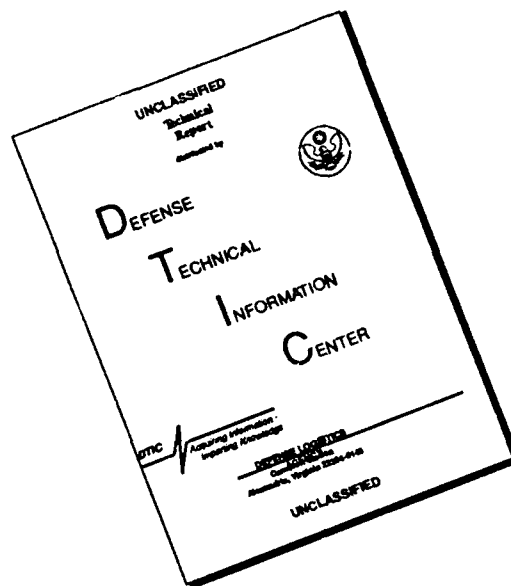
Reproduction of all or part of this report is authorized.

Released by:

  
Richard L. Ester, for

Dean of Faculty and Graduate Studies

# DISCLAIMER NOTICE



THIS DOCUMENT IS BEST QUALITY AVAILABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

Unclassified

security classification of this page

## REPORT DOCUMENTATION PAGE

1a Report Security Classification <b>Unclassified</b>		1b Restrictive Markings	
2a Security Classification Authority		3 Distribution Availability of Report <b>Approved for public release; distribution is unlimited.</b>	
2b Declassification Downgrading Schedule			
4 Performing Organization Report Number(s)		5 Monitoring Organization Report Number(s)	
6a Name of Performing Organization <b>Naval Postgraduate School</b>	6b Office Symbol (if applicable) <b>EC</b>	7a Name of Monitoring Organization <b>Office of the Chief of Naval Operations (OP-03B)</b>	
6c Address (city, state, and ZIP code) <b>Monterey, CA 93943-5000</b>		7b Address (city, state, and ZIP code) <b>Pentagon, Washington, D.C. 20350</b>	
8a Name of Funding Sponsoring Organization <b>Naval Postgraduate School</b>	8b Office Symbol (if applicable) <b>EC</b>	9 Procurement Instrument Identification Number <b>O&amp;MN, Direct Funding</b>	
8c Address (city, state, and ZIP code) <b>Monterey, CA 93943-5000</b>		10 Source of Funding Numbers Program Element No   Project No   Task No   Work Unit Accession No	
11 Title (include security classification) <b>AN ANALYSIS OF MLAYER: A MULTILAYER TROPOSPHERIC PROPAGATION PROGRAM. (Unclassified)</b>			
12 Personal Author(s) <b>Iean-Weng Yeoh</b>			
13a Type of Report <b>Master's Engineer's Thesis</b>	13b Time Covered From   To	14 Date of Report (year, month, day) <b>June 1990</b>	15 Page Count <b>158</b>
16 Supplementary Notation <b>The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.</b>			
17 Cross Codes		18 Subject Terms (continue on reverse if necessary and identify by block number)	
Field	Group	Tropospheric duct propagation, computer program documentation.	
19 Abstract (continue on reverse if necessary and identify by block number) <p>MLAYER, a computer program, was developed by the Naval Ocean Systems Center (NOSC) for calculating the signal levels of electromagnetic waves propagating in a multilayer tropospheric waveguide environment over seawater. The program is an extension of the XWVG which is a trilinear ducting program. Modifications of the XWVG were carried out to handle multilayer tropospheric ducts.</p> <p>A number of modifications and improvements on the program made over the past several years were not documented. A detailed documentation of MLAYER was also not available. The objective of this study is to develop a technical documentation for MLAYER using the program as baseline. The study aims to put together the theoretical formulations (specific to MLAYER) into a complete self-contained document. This is to facilitate potential users with better appreciation of the capabilities, limitations, approximations and assumptions used in the mathematical modelling techniques. As far as possible, the same terminologies and functional variables used by Baumgartner (in the XWVG development) and by Pappert (in the MLAYER development) are adopted to enable one to relate this document to the program. Step-by-step derivation of certain equations was carried out and checked for compatibility with the algorithm in the program. An in-depth scrutiny of each program element was also conducted and a description for each is provided. As a result of a detailed analysis of the respective algorithm in the program, the documentation for the evaluation of the modal function was eventually prepared. Additional materials were gathered from technical reports and papers to supplement the development of this document.</p> <p>The MLAYER supporting programs (Microsoft program maintenance utility "makefiles") were modified to enable the program to run on Microsoft FORTRAN version 5.0. MLAYER was tested and ran successfully on Microsoft FORTRAN version 5.0. and C compilers version 5.0.</p>			
20 Distribution Availability of Abstract <input checked="" type="checkbox"/> unclassified unlimited <input type="checkbox"/> same as report <input type="checkbox"/> DTIC users		21 Abstract Security Classification <b>Unclassified</b>	
22a Name of Responsible Individual <b>Hung-Mou Lee</b>		22b Telephone (include Area code) <b>(408) 646-2846</b>	22c Office Symbol <b>FC 1h</b>

Approved for public release; distribution is unlimited.

An Analysis of MLAYER : A Multilayer Tropospheric  
Propagation Program.

by

Lean-Weng Yeoh  
Ministry of Defence, Singapore  
B.ENG.(Hons), National University of Singapore, 1983  
MSc(EE), National University of Singapore, 1987

Submitted in partial fulfillment of the  
requirements for the degrees of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING  
and  
ELECTRICAL ENGINEER

from the

NAVAL POSTGRADUATE SCHOOL  
June 1990

Author:



Lean-Weng Yeoh

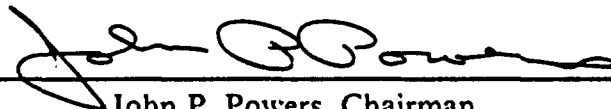
Approved by:



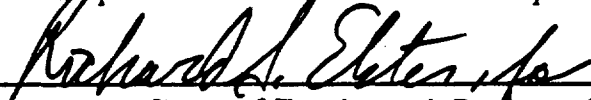
Hung-Mou Lee, Thesis Advisor



Richard W. Adler, Second Reader



John P. Powers, Chairman,  
Department of Electrical and Computer Engineering



Dean of Faculty and Graduate Studies

## ABSTRACT

MLAYER, a computer program, was developed by the Naval Ocean Systems Center (NOSC) for calculating the signal levels of electromagnetic waves propagating in a multilayer tropospheric waveguide environment over seawater. The program is an extension of the XWVG which is a trilinear ducting program. Modifications of the XWVG were carried out to handle multilayer tropospheric ducts.

A number of modifications and improvements on the program made over the past several years were not documented. A detailed documentation of MLAYER was also not available. The objective of this study is to develop a technical documentation for MLAYER using the program as baseline. The study aims to put together the theoretical formulations (specific to MLAYER) into a complete self-contained document. This is to facilitate potential users with better appreciation of the capabilities, limitations, approximations and assumptions used in the mathematical modelling techniques. As far as possible, the same terminologies and functional variables used by Baumgartner (in the XWVG development) and by Pappert (in the MLAYER development) are adopted to enable one to relate this document to the program. Step-by-step derivation of certain equations was carried out and checked for compatibility with the algorithm in the program. An in-depth scrutiny of each program element was also conducted and a description for each is provided. As a result of a detailed analysis of the respective algorithm in the program, the documentation for the evaluation of the modal function was eventually prepared. Additional materials were gathered from technical reports and papers to supplement the development of this document.

The MLAYER supporting programs (Microsoft program maintenance utility "makefiles") were modified to enable the program to run on Microsoft FORTRAN version 5.0. MLAYER was tested and ran successfully on Microsoft FORTRAN version 5.0. and C compilers version 5.0.



Accession For	
NTIS GRA&I	<input checked="checked" type="checkbox"/>
DTIC TAB	<input type="checkbox"/>
Unannounced	<input type="checkbox"/>
Justification	
By	
Distribution/	
Availability Codes	
Dist	Avail and/or Special
A-1	23

## Table of Contents

I.	INTRODUCTION .....	1
II.	MATHEMATICAL FORMULATION OF THE MODEL .....	5
A.	HORIZONTALLY POLARIZED WAVE PROPAGATION .....	5
B.	VERTICALLY POLARIZED WAVE PROPAGATION .....	22
C.	DETERMINATION OF THE HEIGHT GAIN FUNCTION .....	23
D.	TEST FOR EVANESCENCE .....	27
E.	EVALUATION OF THE MODAL FUNCTION .....	32
F.	EFFECTS OF SURFACE ROUGHNESS .....	34
G.	COMPLEX INDEX OF REFRACTION OF SEA WATER .....	42
H.	ATMOSPHERIC ABSORPTION .....	45
III.	SHELLMAN - MORFITT ROOT FINDING ROUTINE .....	51
A.	GENERAL OUTLINE .....	51
B.	ROOT SEARCH PROCEDURE .....	54
C.	DETERMINATION OF THE COORDINATES OF ROOTS .....	61
D.	SOLUTION OF THE QUADRATIC EQUATION .....	67
E.	NEWTON RAPHSON ITERATION .....	69
IV.	SOLUTION OF STOKES' DIFFERENTIAL EQUATION .....	70
A.	STOKES' EQUATION .....	70
B.	EVALUATION OF AIRY FUNCTIONS .....	70
1.	General Outline .....	70
2.	Evaluation by Taylor Series Expansion .....	71
3.	Evaluation by Generalised Gaussian Quadrature Method .....	73
C.	EXTENDED COMPLEX ARITHMETIC .....	76
V.	DESCRIPTION OF MLAYER PROGRAM ELEMENTS .....	78
A.	MAIN .....	78
1.	Description .....	78
2.	Inputs .....	79

3.	Outputs .....	80
4.	Calling subroutine .....	81
5.	Subroutines called .....	81
6.	Common block areas .....	81
B.	SUBROUTINE ABCOEF .....	81
1.	Description .....	81
2.	Calling statement .....	82
3.	Inputs .....	82
4.	Outputs .....	82
5.	Calling program element .....	83
6.	Subroutines called .....	83
7.	Common block areas .....	83
C.	SUBROUTINE ADDX .....	83
1.	Description .....	83
2.	Calling statement .....	83
3.	Inputs .....	84
4.	Outputs .....	84
5.	Calling program element .....	84
6.	Subroutine called .....	84
7.	Common block area .....	84
D.	SUBROUTINE AO2H2O .....	84
1.	Description .....	84
2.	Calling statement .....	84
3.	Inputs .....	84
4.	Output .....	85
5.	Calling program element .....	85
6.	Subroutine called .....	85
7.	Common block area .....	85
E.	SUBROUTINE CHKMOD .....	85
1.	Description .....	85
2.	Calling statement .....	85
3.	Inputs .....	85
4.	Outputs .....	85
5.	Calling program element .....	86
6.	Subroutine called .....	86



7.	Common block area	86
F.	SUBROUTINE DETR	86
1.	Description	86
2.	Calling statement	86
3.	Inputs	86
4.	Outputs	87
5.	Calling program element	87
6.	Subroutine called	87
7.	Common block area	87
G.	SUBROUTINE DIORIZ	87
1.	Description	87
2.	Calling statement	87
3.	Inputs	87
4.	Output	87
5.	Calling program element	88
6.	Subroutine called	88
7.	Common block area	88
H.	SUBROUTINE DXDETR	88
1.	Description	88
2.	Calling statement	88
3.	Inputs	88
4.	Outputs	89
5.	Calling program element	90
6.	Subroutine called	90
7.	Common block area	90
I.	SUBROUTINE FCTVLX	90
1.	Description	90
2.	Calling statement	90
3.	Input	90
4.	Outputs	90
5.	Calling program element	91
6.	Subroutines called	91
7.	Common block areas	91
J.	SUBROUTINE FDFDTX	91
1.	Description	91

2.	Calling statement .....	91
3.	Input .....	91
4.	Outputs .....	91
5.	Calling program element .....	92
6.	Subroutines called .....	92
7.	Common block areas .....	92
K.	SUBROUTINE FINDFX .....	92
1.	Description .....	92
2.	Calling statement .....	92
3.	Inputs .....	92
4.	Outputs .....	93
5.	Calling program element .....	93
6.	Subroutine called .....	93
7.	Common block area .....	93
L.	SUBROUTINE FNDMOD .....	93
1.	Description .....	93
2.	Calling subroutine .....	94
3.	Inputs .....	94
4.	Outputs .....	94
5.	Calling program element .....	95
6.	Subroutine called .....	95
7.	Common block areas .....	95
M.	SUBROUTINE FZEROX .....	95
1.	Description .....	95
2.	Calling statement .....	95
3.	Inputs .....	95
4.	Outputs .....	96
5.	Calling program element .....	96
6.	Subroutines called .....	96
7.	Common block areas .....	96
N.	SUBROUTINE HTGAIN .....	96
1.	Description .....	96
2.	Calling Statement .....	96
3.	Inputs .....	96
4.	Outputs .....	97

5.	Calling program element .....	97
6.	Subroutines called .....	97
7.	Common block areas .....	97
O.	SUBROUTINE MODSUM .....	97
1.	Description .....	97
2.	Calling statement .....	97
3.	Inputs .....	97
4.	Outputs .....	98
5.	Calling program element .....	98
6.	Subroutines called .....	98
7.	Common block areas .....	98
P.	SUBROUTINE NOMSIIX .....	98
1.	Description .....	98
2.	Calling statement .....	98
3.	Inputs .....	98
4.	Output .....	99
5.	Calling program element .....	99
6.	Subroutine called .....	99
7.	Common block area .....	99
Q.	SUBROUTINE NORME .....	99
1.	Description .....	99
2.	Calling statement .....	99
3.	Inputs .....	99
4.	Outputs .....	99
5.	Calling program elements .....	99
6.	Subroutine called .....	100
7.	Common block area .....	100
R.	SUBROUTINE NORMRE .....	100
1.	Description .....	100
2.	Calling statement .....	100
3.	Inputs .....	100
4.	Outputs .....	100
5.	Calling program element .....	100
6.	Subroutine called .....	100
7.	Common block area .....	100

S.	SUBROUTINE QUAD .....	101
1.	Description .....	101
2.	Calling statement .....	101
3.	Inputs .....	101
4.	Outputs .....	101
5.	Calling program element .....	101
6.	Subroutine called .....	101
7.	Common block area .....	101
T.	SUBROUTINE SEAH2O .....	101
1.	Description .....	101
2.	Calling statement .....	102
3.	Inputs .....	102
4.	Outputs .....	102
5.	Calling program element .....	102
6.	Subroutine called .....	102
7.	Common block area .....	102
U.	SUBROUTINE SURF .....	102
1.	Description .....	102
2.	Calling statement .....	102
3.	Input .....	102
4.	Outputs .....	102
5.	Calling program elements .....	103
6.	Subroutines called .....	103
7.	Common block areas .....	103
V.	SUBROUTINE XAINEG .....	103
1.	Description .....	103
2.	Calling statement .....	103
3.	Input .....	103
4.	Outputs .....	104
5.	Calling program element .....	104
6.	Subroutines called .....	104
7.	Common block area .....	104
W.	SUBROUTINE XAIPOS .....	104
1.	Description .....	104
2.	Calling statement .....	104

3.	Input .....	104
4.	Outputs .....	105
5.	Calling program element .....	105
6.	Subroutines called .....	105
7.	Common block area .....	105
X.	SUBROUTINE XCDAI .....	105
1.	Description .....	105
2.	Calling statement .....	106
3.	Input .....	106
4.	Outputs .....	106
5.	Calling program element .....	107
6.	Subroutines called .....	107
7.	Common block area .....	107
Y.	SUBROUTINE XCDAIG .....	107
1.	Description .....	107
2.	Calling statement .....	107
3.	Inputs .....	107
4.	Outputs .....	108
5.	Calling program elements .....	108
6.	Subroutine called .....	108
7.	Common block area .....	108
Z.	SUBROUTINE XCDAIT .....	108
1.	Description .....	108
2.	Calling statement .....	108
3.	Input .....	108
4.	Outputs .....	108
5.	Calling program elements .....	109
6.	Subroutine called .....	109
7.	Common block area .....	109
AA.	FUNCTION PROGRAMS .....	109
1.	Function casin .....	109
2.	Function csech2 .....	109
3.	Function ctanh .....	109
4.	Function ibstrip .....	110
AB.	MLAYER SUPPORTING PROGRAMS .....	110

1. Mlayer.mak .....	111
2. Wvgclean.mak .....	111
3. Wvgstrip.mak .....	111
4. Wvgclean.c .....	111
5. Wvgstrip.for .....	111
6. Wvgstdin.for .....	112
 VI. DISCUSSIONS AND RECOMMENDATIONS .....	113
A. ASSESSMENT OF MLAYER .....	113
B. SURFACE ROUGHNESS MODEL .....	113
C. LATERAL HOMOGENEITY OF REFRACTIVITY PROFILE .....	114
D. FIELD CONTRIBUTION FROM THE SOURCE .....	114
 VII. CONCLUSION .....	115
 APPENDIX A. REPRESENTATION OF $k_1(q)$ AND $k_2(q)$ IN TERMS OF MODIFIED HANKEL FUNCTION OF ORDER ONE-THIRD .....	116
 APPENDIX B. SAMPLE PRINTOUT OF INPUT AND OUTPUT FILES ..	121
 LIST OF REFERENCES .....	139
 INITIAL DISTRIBUTION LIST .....	141

## List of Tables

Table 1.	SPECTROSCOPIC COEFFICIENTS FOR OXYGEN ( $a_1 - a_3$ ) AND WATER ( $b_1 - b_3$ ) [FROM REF. 11] .....	49
Table 2.	NUMBER OF TERMS REQUIRED IN QUADRATURE FORMULA [AFTER REF. 16] .....	75
Table 3.	4-TERM GENERALISED GAUSSIAN INTEGRATION FOR AIRY FUNCTIONS [FROM REF. 16] .....	75
Table 4.	2-TERM GENERALISED GAUSSIAN INTEGRATION FOR AIRY FUNCTIONS [FROM REF. 16] .....	76

## List of Figures

Figure 1.	Multiple piecewise linear modified refractivity profile	3
Figure 2.	Contour of integration of Equation 11 [After Ref. 1]	6
Figure 3.	Branch of $\sqrt{n_z^2 - \beta^2}$	9
Figure 4.	Effects of surface roughness modelled as layer 0 [From Ref. 1]	35
Figure 5.	Constant Phase Line of $f(z)$ [From Ref. 1]	53
Figure 6.	Illustration of search rectangle with mesh grid [From Ref. 1]	56
Figure 7.	Local mesh coordinate system [From Ref. 1]	57
Figure 8.	Crossing of two $\text{Im}(f) = 0$ at the same edge of the mesh square	59
Figure 9.	Two zeros found on the same phase line	60
Figure 10.	$z_r$ is a proper solution of $f(z) = 0$ [After Ref. 14]	65
Figure 11.	$z_r$ is not a proper solution of $f(z) = 0$ [After Ref. 14]	66



## ACKNOWLEDGEMENTS

The author wishes to express his sincere gratitude to Professor Hung-Mou Lee for his professional guidance, invaluable advice and moral support throughout the thesis preparation period. Credit is also given to Dr. Richard A. Pappert and Mr. Ken Anderson of NOSC for making available the MLAYER program, technical reports and papers relevant to MLAYER.

Finally, special thanks are due to Professors Richard W. Adler and John P. Powers for their review and constructive suggestions to the thesis work.

## I. INTRODUCTION

Propagation of radio waves through the troposphere is strongly affected by the complex nature of this propagation medium. The effects on radio wave propagation are due to changes in the refractive index with height, absorption by the atmosphere and its constituent gases, attenuation and scattering due to climatic conditions.

Although weather conditions could result in attenuation of radio wave transmission through the absorption of energy by gases, water vapour or scattering by rain, certain meteorological conditions can lead to an enhancement of signal strength. This peculiar phenomenon known as ducting is due to the inversions of the refractive index profile. Propagation in a tropospheric duct can result in an unusual increase in range.

To predict the anomalous propagation effects, knowledge of the characteristics of the refractive index and its effects on propagated fields is required. In the mathematical formulation of the model, the concept of modified index of refraction is used to describe wave propagation over a flat earth.

In a uniform atmosphere with constant index of refraction, electromagnetic waves travel in straight lines over the curved surface of the earth. This situation is unaltered if the earth is considered as flat and the rays curved, provided the relative curvature between the earth and the rays is preserved. To account for the upward bending of rays over the flat earth, the constant index of refraction is transformed to a modified index which increases with height and is given by

$$m(z) = n + \frac{z}{a} \quad , \quad (1)$$

where  $n$  is the index of refraction,  $z$  is the height above the surface of the earth and  $a$  is the radius of the earth. Equation 1 is still valid in a stratified atmosphere where the index of refraction is a function of height. In this case,  $n$  is replaced by  $n(z)$  in Equation 1.

The modified index concept allows one to transform a spherically stratified refractive structure into a planar layer by preserving the relative curvature between the normal of the radio-wavefront and the earth's surface. Therefore wave propagation over a spherical earth can be reduced to that over a flat earth if the index of refraction is replaced by the modified index of refraction. For  $z/a \ll 1$ , the angle of inclination of a ray

at any point in space and the distance (optical length) along the ray path will remain invariant in the transformation.

The flat-earth approximation is adopted in MLAYER [Ref. 1 and 2]. This approximation considerably simplifies the mathematical solution of the problem. The accuracy of the flat-earth approximation was investigated by Pekeris [Ref. 3]. It was found that for a standard atmosphere, the fractional error in the height gain function is approximately

$$e = \frac{2\pi\sqrt{6}}{5} \frac{z^{5/2}}{\lambda a^{3/2}} \quad (2)$$

Equation 2 shows that the flat-earth approximation will break down at high frequency and at great height above the earth's surface. For example, a 10 GHz signal at a height of 2000 m will give a fractional error of approximately 114%. The work of Pekeris also indicates that the flat-earth approximation is valid to within two percent for ranges up to about one-half of the radius of the earth [Ref. 3].

As typical values of  $m(z)$  differ from unity by about 300 parts per million, it is sometimes more convenient to introduce the modified refractivity to describe the phenomena involved in the formation of tropospheric ducts. The modified refractivity is defined as

$$M(z) = [m(z) - 1] \times 10^6 \quad (3)$$

In this program, the modified refractivity profile of the troposphere is modelled as a series of continuous piecewise linear segments (see Figure 1).

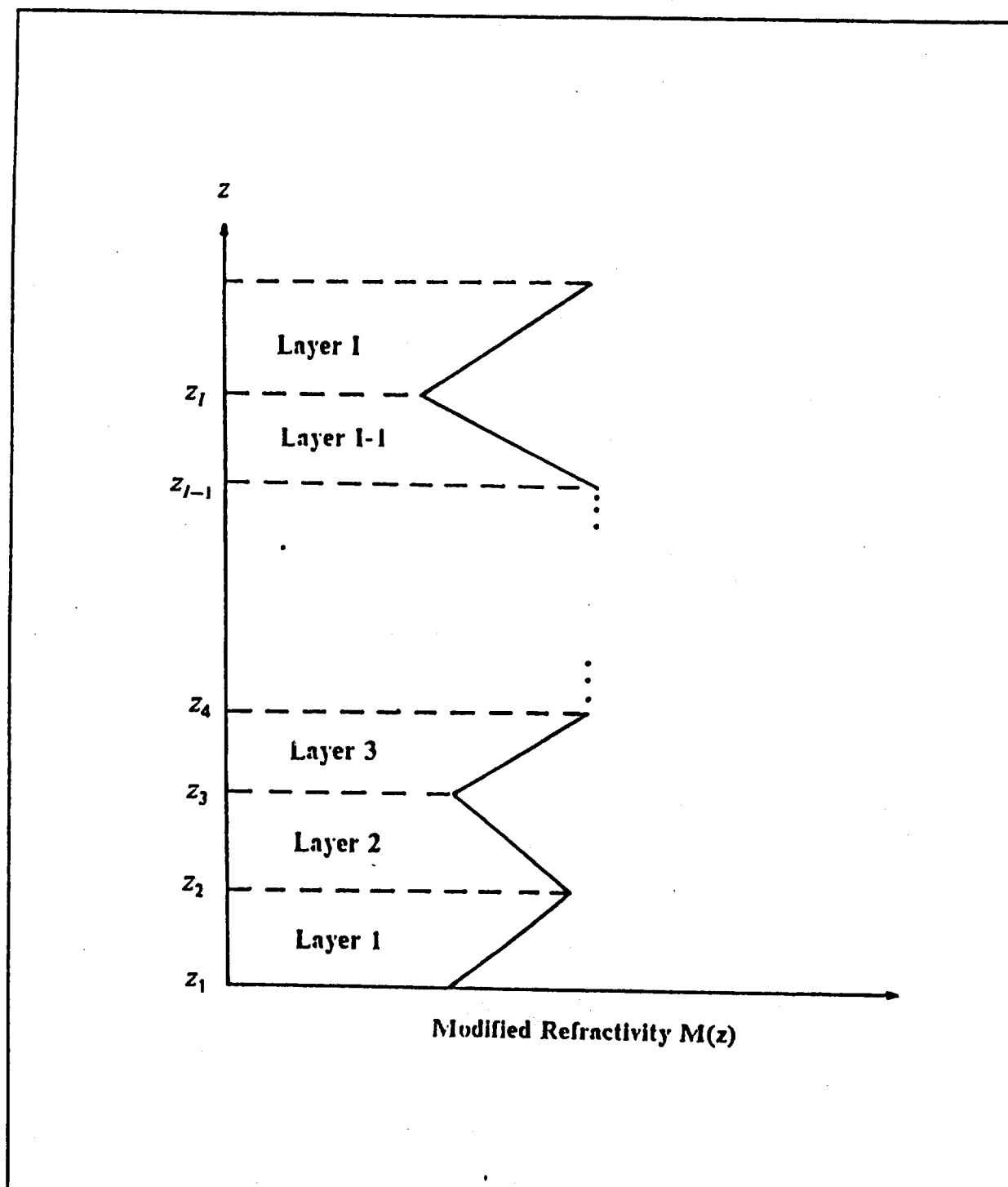
The linear approximation simplifies mathematical manipulations to a large extent. In general, any complicated profile can be approximated by several linear segments and the accuracy of the modelling can be improved by increasing the number of linear segments and reducing the thickness of each layer.

If  $M_i(z)$  is the value of the modified refractivity in the  $i^{\text{th}}$  layer, then for a linear modified refractivity profile,

$$M_i(z) = M(z_i) + \frac{dM_i}{dz} (z - z_i) ; \quad (4)$$

$$z_i \leq z \leq z_{i+1} .$$

From Equation 3, the square of the modified index of refraction is



**Figure 1. Multiple piecewise linear modified refractivity profile**

$$m_i^2(z) \simeq 1 + 2 \times 10^{-6} M_l(z) - j\eta_l, \quad (5)$$

where  $\eta_l$  is introduced to account for attenuation of the signal due to the absorption by atmospheric gases. The derivative of  $m_i^2(z)$  is given by

$$\begin{aligned} \frac{d}{dz} \{m_i^2(z)\} &\simeq 2 \times 10^{-6} \frac{dM_l(z)}{dz} - j \frac{d\eta_l}{dz} \\ &\equiv \alpha_l. \end{aligned} \quad (6)$$

Equations 4, 5 and 6 will be used in the mathematical model to represent the characteristics of the propagation medium.

MLAYER also incorporates in the solution of the modal function, the effects of surface roughness, atmospheric absorption and the variation of the complex index of refraction of seawater as a function of temperature, salinity and frequency.

The program uses the Shellman-Morfitt complex root-finding routine to locate the modes that propagate in the tropospheric ducts. This routine attempts to find all waveguide modes with attenuation rates below a user-specified value.

Overflow problems commonly encountered in the evaluation of the modal function are avoided by performing numerical computations with extended complex arithmetic.

## II. MATHEMATICAL FORMULATION OF THE MODEL

### A. HORIZONTALLY POLARIZED WAVE PROPAGATION

In formulating the problem, the flat-earth approximation with modified index of refraction is adopted. The cylindrical coordinate system,  $(r, \phi, z)$ , is used. The vertical height,  $z$ , is measured from the surface of the flat-earth.

A horizontally polarized source may be approximated as a radiating magnetic dipole oriented in the  $z$  direction and located at  $r=0$  and  $z = z_T$ . The magnetic Hertz vector associated with the dipole is given by

$$\vec{\Pi}(r, z) = \Pi(r, z) \hat{z} , \quad (7)$$

where  $\Pi(r, z)$  satisfies the following equation:

$$\nabla^2 \Pi(r, z) + k^2 m^2(z) \Pi(r, z) = -4\pi \{\delta(x) \delta(y) \delta(z - z_T)\} , \quad (8)$$

where  $k$  is the wave number in free space,  $\delta(\cdot)$  is the Dirac delta function and  $z_T$  is the location of the source. The right hand side of Equation 8 corresponds to a point source of strength  $-4\pi$  located at  $r=0$  and  $z = z_T$ . The time dependence  $e^{j\omega t}$  is assumed throughout this thesis.

The electric and magnetic field can be expressed in terms of the Hertz vector as

$$\vec{E} = -j\omega\mu_0 \vec{\nabla} \times \vec{\Pi} \quad (9)$$

and

$$\vec{H} = \vec{\nabla} \times \vec{\nabla} \times \vec{\Pi} , \quad (10)$$

where  $\mu_0$  is the free space permeability. Solution to Equation 8 can be represented as a contour integral:

$$\Pi(r, z) = \frac{1}{2} \int_C \rho d\rho H_0^{(2)}(r\rho) f(\rho, z) , \quad (11)$$

where  $H_0^{(2)}(r\rho)$  is the zeroth-order Hankel function of the second kind and  $f(\rho, z)$  is the height gain function. The contour of integration,  $C$ , as shown in Figure 2 can be deformed into  $C_1$  and  $C_2$ , also shown in Figure 2.

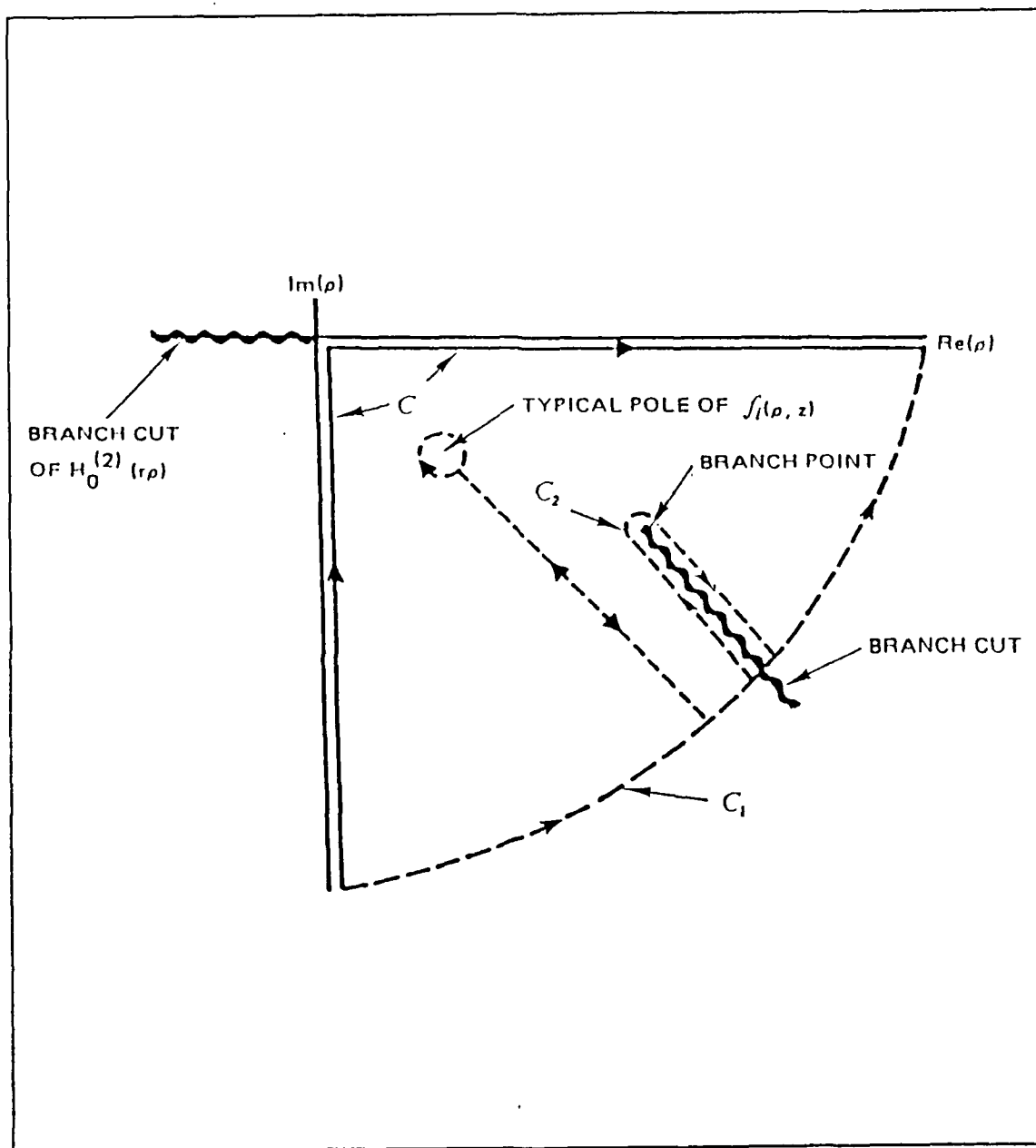


Figure 2. Contour of integration of Equation 11 [After Ref. 1]

The height gain function in the  $i^{\text{th}}$  tropospheric layer, denoted as  $f_i(\rho, z)$ , satisfies the following differential equation:

$$\left\{ \frac{\partial^2}{\partial z^2} + k^2 \left[ m_i^2(z) - \frac{\rho^2}{k^2} \right] \right\} f_i(\rho, z) = -2\pi\delta(z - z_T) . \quad (12)$$

In the ground, the height gain function,  $f_g(\rho, z)$ , is given by the solution of the following equation:

$$\left\{ \frac{\partial^2}{\partial z^2} + k^2 \left[ n_g^2 - \frac{\rho^2}{k^2} \right] \right\} f_g(\rho, z) = 0 , \quad (13)$$

where  $n_g$  is the complex index of refraction of the ground and is related to the relative permittivity,  $\epsilon_g$ , and conductivity,  $\sigma_g$ , of the ground as

$$n_g^2 = \epsilon_g - j \frac{\sigma_g}{\epsilon_0 \omega} . \quad (14)$$

The height gain function,  $f_g(\rho, z)$ , must represent an outgoing exponentially decaying wave as  $z \rightarrow -\infty$ , i.e.,

$$f_g(\rho, z) = A_g e^{j\gamma z} , \quad (15)$$

where

$$\gamma = k \sqrt{n_g^2 - \beta^2} \quad (16)$$

and

$$\beta = \frac{\rho}{k} . \quad (17)$$

For the wave to attenuate into the ground, the branch of the square root in Equation 16 should be chosen such that

$$\text{Im}(\gamma) < 0 . \quad (18)$$

If the ground is not dissipative, the branch point and the integration contour satisfying Equation 18 is shown in Figure 3.



Continuity of the tangential components of  $\vec{E}$  and  $\vec{H}$  at each layer boundary requires that

$$f_g(\rho, 0) = f_1(\rho, 0) , \quad (19)$$

$$\frac{\partial}{\partial z} \{f_g(\rho, 0)\} = \frac{\partial}{\partial z} \{f_1(\rho, 0)\} , \quad (20)$$

$$f_i(\rho, z_{i+1}) = f_{i+1}(\rho, z_{i+1}) ; i = 1 \text{ to } I - 1 \quad (21)$$

and

$$\frac{\partial}{\partial z} \{f_i(\rho, z_{i+1})\} = \frac{\partial}{\partial z} \{f_{i+1}(\rho, z_{i+1})\} ; i = 1 \text{ to } I - 1 . \quad (22)$$

Define

$$q_i = \left[ \frac{k}{\alpha_i} \right]^{2/3} [m^2(z_i) + \alpha_i(z - z_i) - \beta^2] \quad (23)$$

and with a change of variable, the solution to Equation 12 can be expressed as a linear combination of  $k_1(q_i)$  and  $k_2(q_i)$ , satisfying

$$\left[ \frac{\partial^2}{\partial q_i^2} + q_i \right] k_n(q_i) = 0 ; n = 1, 2 . \quad (24)$$

This is Stokes' differential equation. For  $i \neq I$

$$f_i(\rho, z) = B_i(\rho) [A_i(\rho)k_1(q_i) + k_2(q_i)] , \quad (25)$$

where  $k_1(q_i)$  and  $k_2(q_i)$  are proportional to the Airy functions

$$k_1(q_i) \sim Ai(-q_i e^{j2\pi/3}) \quad (26)$$

and

$$k_2(q_i) \sim Ai(-q_i) . \quad (27)$$

Note that the  $A_i$ 's and  $B_i$ 's are different from those in Reference 2 where the height gain function is expressed as  $B_i(k_1 + A_i k_2)$ . Equation 25 corresponds to that implemented in the program.

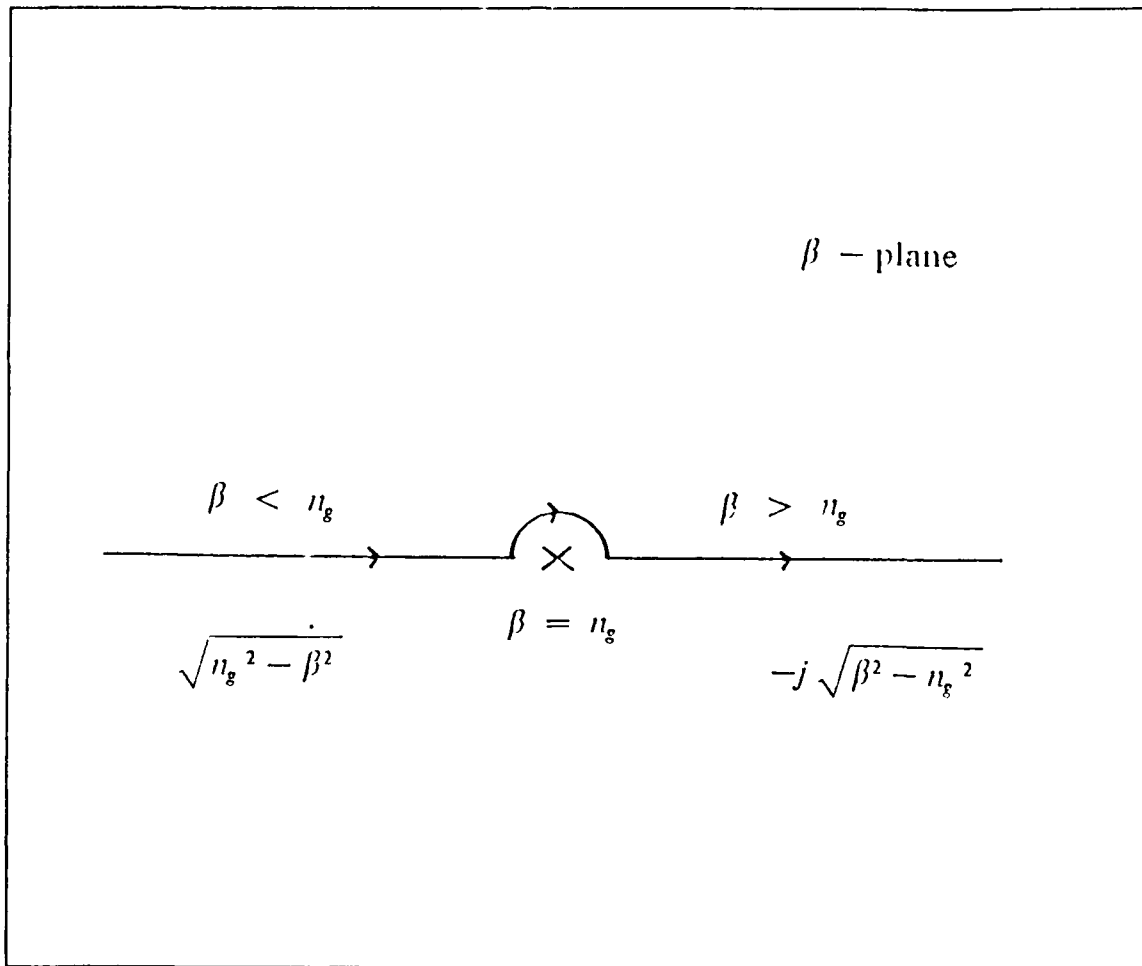


Figure 3. Branch of  $\sqrt{n_g^2 - \beta^2}$

The constants of proportionality of Equations 26 and 27 are of no consequence as they affect only  $B_1(\rho)$  and  $A_1(\rho)$  of Equation 25 but not  $f_1(\rho, z)$ . The height gain functions,  $f_1(\rho, z)$ , used in the computations of the mode sum and path loss are normalized as in Equation 77. The solution of Stokes' equation is discussed in Chapter IV.

For the highest layer,  $i=I$ , the height gain function is given by

$$f_I(\rho, z) = B_I(\rho) h_2(q_I) \quad , \quad (28)$$

where  $h_2(q_I)$  is the modified Hankel function of order one-third and is proportional to the Airy function as

$$h_2(q_I) \sim Ai(-q_I e^{-j2\pi/3}) \quad . \quad (29)$$

As in Equations 26 and 27, the constant of proportionality of Equation 29 is also omitted in the program.

Equation 25 represents the solution of  $f(\rho, z)$  for every layer except the one in which the source is located. In the layer where the source is located, the height gain function is given by

$$f_i(\rho, z) = B_i(\rho) [A_i(\rho) k_1(q_i) + k_2(q_i)] + f^{(s)}(\rho, q_i) \quad (30)$$

where  $f^{(s)}(\rho, q)$  is the particular solution of the inhomogenous Equation 12. Therefore the complete solutions of  $f_i(\rho, z)$  are given by

$$f_i(\rho, z) = B_i(\rho) [A_i(\rho) k_1(q_i) + k_2(q_i)] + f^{(s)}(\rho, q_i) \delta_{iM} \quad ; \quad i = 1 \text{ to } I-1 \quad (31)$$

and

$$f_i(\rho, z) = B_i(\rho) h_2(q_i) + f^{(s)}(\rho, q_i) \delta_{iM} \quad ; \quad i = I \quad (32)$$

where

$$\delta_{iM} = \begin{cases} 1 & i = M \\ 0 & i \neq M \end{cases} \quad (33)$$

It is assumed that the source is located in the  $M^{\text{th}}$  layer.

Continuity of  $f(\rho, z)$  and  $\frac{\partial}{\partial z} f(\rho, z)$  at the ground level requires that

$$A_g = B_1 [A_1 k_1(q_{1,1}) + k_2(q_{1,1})] + f^{(s)}(\rho, q_{1,1}) \delta_{1M} \quad (34)$$

and

$$j\gamma A_g = B_1 [A_1 k'_1(q_{1,1}) + k'_2(q_{1,1})] \left[ \frac{k}{\alpha_1} \right]^{2/3} \alpha_1 + \frac{\partial}{\partial z} [f^{(s)}(\rho, q_{1,1})] \delta_{1M} \quad (35)$$

From Equations 34 and 35,

$$\begin{aligned} & \left\{ \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 k'_1(q_{1,1}) - j\gamma k_1(q_{1,1}) \right\} B_1 A_1 + \left\{ \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 k'_2(q_{1,1}) - j\gamma k_2(q_{1,1}) \right\} B_1 \\ & = j\gamma f^{(s)}(\rho, q_{1,1}) \delta_{1M} - \frac{\partial}{\partial z} [f^{(s)}(\rho, q_{1,1})] \delta_{1M} \end{aligned} \quad (36)$$

Similarly, at layer boundaries,  $i = 1$  to  $I-2$ ,

$$\begin{aligned}
& B_i [A_i k_1(q_{i,i+1}) + k_2(q_{i,i+1})] + f^{(s)}(\rho, q_{i,i+1}) \delta_{i,M} \\
& = B_{i+1} [A_{i+1} k_1(q_{i+1,i+1}) + k_2(q_{i+1,i+1})] \\
& + f^{(s)}(\rho, q_{i+1,i+1}) \delta_{i+1,M}
\end{aligned} \tag{37}$$

and

$$\begin{aligned}
& B_i [A_i k'_1(q_{i,i+1}) + k'_2(q_{i,i+1})] \left[ \frac{k}{\alpha_i} \right]^{2/3} \alpha_i \\
& + \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{i,i+1})] \delta_{i,M} \\
& = B_{i+1} [A_{i+1} k'_1(q_{i+1,i+1}) + k'_2(q_{i+1,i+1})] \left[ \frac{k}{\alpha_{i+1}} \right]^{2/3} \alpha_{i+1} \\
& + \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{i+1,i+1})] \delta_{i+1,M} .
\end{aligned} \tag{38}$$

From Equation 37,

$$\begin{aligned}
& k_1(q_{i,i+1}) B_i A_i + k_2(q_{i,i+1}) B_i \\
& - k_1(q_{i+1,i+1}) B_{i+1} A_{i+1} - k_2(q_{i+1,i+1}) B_{i+1} \\
& = f^{(s)}(\rho, q_{i+1,i+1}) \delta_{i+1,M} - f^{(s)}(\rho, q_{i,i+1}) \delta_{i,M} .
\end{aligned} \tag{39}$$

From Equation 38,

$$\begin{aligned}
& k'_1(q_{i,i+1}) \left( \frac{k}{\alpha_i} \right)^{2/3} \alpha_i B_i A_i + k'_2(q_{i,i+1}) \left( \frac{k}{\alpha_i} \right)^{2/3} \alpha_i B_i \\
& - k'_1(q_{i+1,i+1}) \left( \frac{k}{\alpha_{i+1}} \right)^{2/3} \alpha_{i+1} B_{i+1} A_{i+1} \\
& - k'_2(q_{i+1,i+1}) \left( \frac{k}{\alpha_{i+1}} \right)^{2/3} \alpha_{i+1} B_{i+1} \\
& = \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{i+1,i+1})] \delta_{i+1,M} \\
& - \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{i,i+1})] \delta_{i,M} .
\end{aligned} \tag{40}$$

For  $i = 1-1$ ,

$$\begin{aligned} B_{I-1} [A_{I-1} k_1(q_{I-1,I}) + k_2(q_{I-1,I})] + f^{(s)}(\rho, q_{I-1,I}) \delta_{I-1,M} \\ = B_I h_2(q_{I,I}) + f^{(s)}(\rho, q_{I,I}) \delta_{I,M} \end{aligned} \quad (41)$$

and

$$\begin{aligned} B_{I-1} [A_{I-1} k'_1(q_{I-1,I}) + k'_2(q_{I-1,I})] \left( \frac{k}{\alpha_{I-1}} \right)^{2/3} \alpha_{I-1} \\ + \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{I-1,I})] \delta_{I-1,M} \\ = B_I h'_2(q_{I,I}) \left( \frac{k}{\alpha_I} \right)^{2/3} \alpha_I + \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{I,I})] \delta_{I,M} . \end{aligned} \quad (42)$$

From Equation 41,

$$\begin{aligned} k_1(q_{I-1,I}) B_{I-1} A_{I-1} + k_2(q_{I-1,I}) B_{I-1} - h_2(q_{I,I}) B_I \\ = f^{(s)}(\rho, q_{I,I}) \delta_{I,M} - f^{(s)}(\rho, q_{I-1,I}) \delta_{I-1,M} . \end{aligned} \quad (43)$$

From Equation 42,

$$\begin{aligned} k'_1(q_{I-1,I}) \left( \frac{k}{\alpha_{I-1}} \right)^{2/3} \alpha_{I-1} B_{I-1} A_{I-1} + k'_2(q_{I-1,I}) \left( \frac{k}{\alpha_{I-1}} \right)^{2/3} \alpha_{I-1} B_{I-1} \\ - h'_2(q_{I,I}) \left( \frac{k}{\alpha_I} \right)^{2/3} \alpha_I B_I \\ = \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{I,I})] \delta_{I,M} - \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{I-1,I})] \delta_{I-1,M} . \end{aligned} \quad (44)$$

In the above equations,  $q_{i,k}$  is defined as

$$q_{i,k} = q_i(z = z_k) . \quad (45)$$

Equations 36, 39, 40, 43 and 44 can be written in matrix notation as:

$$\underline{\Delta} \underline{\xi} = \underline{\psi} , \quad (46)$$

$$\Delta =$$

(47)

$$\underline{\rho^{(n)}} = \begin{bmatrix} B_1 A_1 \\ B_1 \\ B_2 A_2 \\ B_2 \\ \vdots \\ \vdots \\ B_{I-1} A_{I-1} \\ B_{I-1} \\ B_I \end{bmatrix} \quad (48)$$

and

$$\underline{\psi} = \begin{bmatrix} \psi_1 \\ \psi_2 \\ \vdots \\ \vdots \\ \psi_I \end{bmatrix} . \quad (49)$$

The elements of the matrix,  $\underline{\Delta}$ , are

$$\Delta_{1,1} = \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 k'_1(q_{1,1}) - j\gamma k_1(q_{1,1}) , \quad (50)$$

$$\Delta_{1,2} = \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 k'_2(q_{1,1}) - j\gamma k_2(q_{1,1}) , \quad (51)$$

$$\Delta_{2,1} = k_1(q_{1,2}) , \quad (52)$$

$$\Delta_{2,2} = k_2(q_{1,2}) , \quad (53)$$

$$\Delta_{2,3} = -k_1(q_{2,2}) . \quad (54)$$

$$\Delta_{2,1} = -k_2(q_{2,2}) \quad , \quad (55)$$

$$\Delta_{3,1} = k'_1(q_{1,2}) \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 \quad , \quad (56)$$

$$\Delta_{3,2} = k'_2(q_{1,2}) \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 \quad , \quad (57)$$

$$\Delta_{3,3} = -k'_1(q_{2,2}) \left( \frac{k}{\alpha_2} \right)^{2/3} \alpha_2 \quad , \quad (58)$$

$$\Delta_{3,4} = k'_2(q_{2,2}) \left( \frac{k}{\alpha_2} \right)^{2/3} \alpha_2 \quad , \quad (59)$$

$$\vdots$$

$$\vdots$$

$$\Delta_{2I-2,2I-3} = k_1(q_{I-1,I}) \quad , \quad (60)$$

$$\Delta_{2I-2,2I-2} = k_2(q_{I-1,I}) \quad , \quad (61)$$

$$\Delta_{2I-2,2I-1} = -h_2(q_{I,I}) \quad , \quad (62)$$

$$\Delta_{2I-1,2I-3} = k'_1(q_{I-1,I}) \left( \frac{k}{\alpha_{I-1}} \right)^{2/3} \alpha_{I-1} \quad , \quad (63)$$

$$\Delta_{2I-1,2I-2} = k'_2(q_{I-1,I}) \left( \frac{k}{\alpha_{I-1}} \right)^{2/3} \alpha_{I-1} \quad (64)$$



and

$$\Delta_{2I-1,2I-1} = -h'_2(q_{I,I}) \left( \frac{k}{z_I} \right)^{2/3} \alpha_I . \quad (65)$$

The elements of the column vector,  $\underline{\psi}$ , are

$$\psi_1 = j f^{(s)}(\rho, q_{1,1}) \delta_{1M} - \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{1,1})] \delta_{1M} , \quad (66)$$

$$\psi_2 = f^{(s)}(\rho, q_{2,2}) \delta_{2M} - f^{(s)}(\rho, q_{1,2}) \delta_{1M} . \quad (67)$$

$$\psi_3 = \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{2,2})] \delta_{2M} - \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{1,2})] \delta_{1M} . \quad (68)$$

$\vdots$

$$\vdots$$

$$\psi_{I-1} = f^{(s)}(\rho, q_{I,I}) \delta_{IM} - f^{(s)}(\rho, q_{I-1,I}) \delta_{I-1,M} \quad (69)$$

and

$$\psi_I = \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{I,I})] \delta_{IM} - \frac{\hat{c}}{\hat{c}z} [f^{(s)}(\rho, q_{I-1,I})] \delta_{I-1,M} . \quad (70)$$

To determine the modes that propagate in the tropospheric ducts, the roots of

$$\|\underline{\Delta}\| = 0 \quad (71)$$

have to be located. This is discussed in Chapter III. The symbol,  $\|\underline{\Delta}\|$ , denotes the determinant of  $\underline{\Delta}$ .

From Equation 11, the solution of Equation 8 in the  $i^{\text{th}}$  layer is

$$\Pi(r, z) = \frac{1}{2} \int_C \rho d\rho H_0^{(2)}(r\rho) f_i(\rho, z) . \quad (72)$$

From the theory of residues, Equation 72 can be expressed as

$$\begin{aligned}\Pi(r, z) = & -\frac{2\pi j}{2} \sum_m b_i(m) + \frac{1}{2} \int_{C_1} \rho d\rho H_0^{(2)}(r\rho) f_i(\rho, z) \\ & + \frac{1}{2} \int_{C_2} \rho d\rho H_0^{(2)}(r\rho) f_i(\rho, z) ,\end{aligned}\quad (73)$$

where  $b_i(m)$  are the residues of  $\rho H_0^{(2)}(r\rho) f_i(\rho, z)$  evaluated at the poles of  $f_i(\rho, z)$  in the fourth quadrant. The contour  $C_1$  is the quarter-circle of infinite radius and  $C_2$  is the contour enclosing the branch cut of the radical of Equation 16. Since  $H_0^{(2)}(r\rho)$  decays exponentially in the lower half complex  $\rho$ -plane,

$$\int_{C_1} \rho d\rho H_0^{(2)}(r\rho) f_i(\rho, z) = 0 . \quad (74)$$

For small height,  $z$ , the integral over contour  $C_2$  represents the contribution by the surface wave to the total field. Beyond the horizon, this field is small compared to the total field and can be neglected. Therefore Equation 73 becomes

$$\Pi(r, z) = -\pi j \sum_m b_i(m) , \quad (75)$$

where  $b_i(m)$  is given by

$$b_i(m) = H_0^{(2)}(r\rho_m) g_m(z) g_m(z_T) \quad (76)$$

and  $g_m(z)$  is the normalized height gain function given by [Ref. 2]

$$g_m(z) = \frac{f_m[q(z)]}{\sqrt{\int_0^\infty f_m^2[q(z)] dz + \frac{f_m^2[q(0)]}{2\rho_m} \left( j \frac{d_i}{d\rho} \right)_{\rho=\rho_m}}} , \quad (77)$$

where  $q(z) = q(z)$  for  $z_i \leq z < z_{i+1}$ .

According to the program source code, the integral,  $II = \int_0^\infty f_m^2[q(z)] dz$ , is evaluated by considering that  $f_m[q(z)]$  is a solution of Stokes's equation,

$$\frac{d^2 f_m}{dq^2} + q f_m = 0 . \quad (78)$$

This leads to

$$\begin{aligned}
 \frac{d}{dq} \left[ q f_m^2 + \left( \frac{df_m}{dq} \right)^2 \right] &= f_m^2 + 2q f_m \frac{df_m}{dq} + 2 \frac{df_m}{dq} \cdot \frac{d^2 f_m}{dq^2} \\
 &= f_m^2 + 2 \frac{df_m}{dq} \left[ \frac{d^2 f_m}{dq^2} + q f_m \right] \\
 &= f_m^2 .
 \end{aligned} \tag{79}$$

Note that Equations 78 and 79 imply that the derivative of  $f_m$  is continuous at  $z_i$ . That is,  $f_m(q)$  does not include the contribution from the last term of Equations 31 and 32. The integral of  $f_m^2[q(z)]$  in the  $i^{\text{th}}$  layer is given by

$$\begin{aligned}
 \int_{z_i}^{z_{i+1}} f_m^2[q(z)] dz &= \frac{1}{q'_i} \int_{q_{i,i}}^{q_{i,i+1}} f_m^2(q_i) dq_i \\
 &= \frac{1}{q'_i} \left\{ q f_m^2(q_i) + \left[ \frac{df_m(q_i)}{dq_i} \right]^2 \right\}_{q_{i,i}}^{q_{i,i+1}} ,
 \end{aligned} \tag{80}$$

where

$$q'_i = \frac{dq_i}{dz} . \tag{81}$$

From Equation (80), it can be seen that the contribution of  $\Delta I_i$  to  $I$  at  $z = z_i$  ( $i \neq 1$ ) is given by

$$\begin{aligned}
 \Delta I_i &= \frac{1}{q'_i} \left\{ q_i f_m^2(q_i) + \left[ \frac{df_m(q_i)}{dq_i} \right]^2 \right\}_{q_{i,i}}^{q_{i,i+1}} \\
 &= \left[ -\frac{q_{i,i}}{q'_i} + \frac{q_{i-1,i}}{q'_{i-1}} \right] f_m^2(q_{i,i}) \\
 &\quad - \frac{1}{q'_i} \left\{ \frac{d}{dq_i} [f_m(q_{i,i})] \right\}^2 + \frac{1}{q'_{i-1}} \left\{ \frac{d}{dq_{i-1}} [f_m(q_{i-1,i})] \right\}^2 .
 \end{aligned} \tag{82}$$

By making use of the fact that  $\frac{df_m}{dz}$  is continuous at the layer boundaries,

$$\frac{d}{dz} [f_m(q_{i-1,i})] = \frac{d}{dz} [f_m(q_{i,i})] . \tag{83}$$

which implies

$$q_{i-1}' \frac{d}{dq_{i-1}} [f_m(q_{i-1,i})] = q_i' \frac{d}{dq_i} [f_m(q_{i,i})] \quad (84)$$

and

$$\left\{ \frac{d}{dq_{i-1}} [f_m(q_{i-1,i})] \right\}^2 = \left( \frac{q_i'}{q_{i-1}'} \right)^2 \left\{ \frac{d}{dq_i} [f_m(q_{i,i})] \right\}^2 \quad (85)$$

Substituting Equation 85 into Equation 82,  $\Delta I_i$  becomes

$$\begin{aligned} \Delta I_i = & \left[ -\frac{q_{i,i}}{q_i'} + \frac{q_{i-1,i}}{q_{i-1}'} \right] f_m^2(q_{i,i}) \\ & + \left[ -\frac{1}{q_i'} + \frac{(q_i')^2}{(q_{i-1}')^3} \right] \left\{ \frac{d}{dq_i} [f_m(q_{i,i})] \right\}^2 \end{aligned} \quad (86)$$

For  $i=1$ ,

$$\Delta I_1 = -\frac{1}{q_1'} q_{1,1} f_m^2(q_{1,1}) - \frac{1}{q_1'} \left\{ \frac{d}{dq_1} [f_m(q_{1,1})] \right\}^2 \quad (87)$$

The integral,  $II$ , is

$$II = \sum_{i=1}^l \Delta I_i \quad (88)$$

The second term of the denominator of Equation 77 can be written as

$$\begin{aligned} \frac{f_m^2[q(0)]}{2\rho_m} j\left(\frac{d_i}{d\rho}\right)_{\rho=\rho_m} &= \frac{f_m^2[q(0)]}{2\rho_m} j\left(\frac{d_i}{dq_1} \cdot \frac{dq_1}{d\rho}\right)_{\rho=\rho_m} \\ &= j \frac{f_m^2[q(0)]}{2\rho_m} \frac{d_i}{dq_1} \left(\frac{k}{\alpha_1}\right)^{2/3} \frac{2\rho_m}{k^2} \\ &= j \frac{f_m^2[q(0)]}{k^2} \left(\frac{k}{\alpha_1}\right)^{2/3} \frac{d_i}{dq_1} \end{aligned} \quad (89)$$

For cases of interest,  $|\rho_m r| \gg 1$ . The Hankel function in Equation 76 can be replaced by its asymptotic approximation:

$$H_0^2(\rho_m r) \sim \left( \frac{2}{\pi \rho_m r} \right)^{1/2} \exp \left( -j \rho_m r + j \frac{\pi}{4} \right) \quad (90)$$

Therefore Equation 75 becomes

$$\begin{aligned} \Pi(r, z) &= -\pi j \sum_m \left( \frac{2}{\pi \rho_m r} \right)^{1/2} \exp \left( -j \rho_m r + j \frac{\pi}{4} \right) g_m(z) g_m(z_T) \\ &= \pi e^{-j\pi/4} \left( \frac{2}{\pi r} \right)^{1/2} e^{j\pi/4} \sum_m \rho_m^{-1/2} g_m(z) g_m(z_T) e^{-j\rho_m r} \\ &= \left( \frac{2\pi}{r} \right)^{1/2} e^{-j\pi/4} \sum_m \rho_m^{-1/2} g_m(z) g_m(z_T) e^{-j\rho_m r} \end{aligned} \quad (91)$$

To allow for spherical spreading, the  $r^{-1/2}$  term in Equation 91 can be replaced by  $[a \sin(\frac{r}{a})]^{-1/2}$ ,  $a$  being the earth's radius. Equation 91 becomes

$$\Pi(r, z) = \left[ \frac{2\pi}{a \sin(\frac{r}{a})} \right]^{1/2} e^{-j\pi/4} \sum_m \rho_m^{-1/2} g_m(z) g_m(z_T) e^{-j\rho_m r} \quad (92)$$

In the cylindrical coordinate system,  $(r, \phi, z)$ , the transverse electric field component,  $E_\phi$ , is given by

$$E_\phi = -\omega \mu_0 \frac{\partial \Pi}{\partial r} \quad (93)$$

Differentiating Equation 92 with respect to  $r$  gives

$$\begin{aligned} \frac{\partial \Pi}{\partial r} &= \left\{ -\left( \frac{2\pi}{a} \right)^{1/2} \left( \frac{\cos(\frac{r}{a})}{2a} \right) \frac{1}{\sqrt{\sin^3(\frac{r}{a})}} - j \left[ \frac{2\pi}{a \sin(\frac{r}{a})} \right]^{1/2} \rho_m \right\} \\ &\quad \times e^{-j\pi/4} \sum_m \rho_m^{-1/2} g_m(z) g_m(z_T) e^{-j\rho_m r} \end{aligned} \quad (94)$$

Neglecting higher order terms of  $\frac{1}{r}$ , Equation 94 becomes

$$\frac{\partial \Pi}{\partial r} = -j \left[ \frac{2\pi}{a \sin(\frac{r}{a})} \right]^{1/2} e^{-j\pi/4} \sum_m \rho_m^{1/2} g_m(z) g_m(z_T) e^{-j\rho_m r} \quad (95)$$

The free-space solution to Equation 8 is given by

$$\Pi_{FS} = \frac{e^{-jk \sqrt{r^2 + z^2}}}{\sqrt{r^2 + z^2}} \simeq \frac{e^{-jkr}}{r} \quad ; \quad r \gg z \quad (96)$$

Thus,

$$\frac{\partial \Pi_{FS}}{\partial r} = \frac{jk}{r} e^{-jkr} - \frac{e^{-jkr}}{r^2} \quad (97)$$

and similarly, the higher order term of  $\frac{1}{r}$  in Equation 97 is neglected, giving

$$\frac{\partial \Pi_{FS}}{\partial r} \simeq \frac{jk}{r} e^{-jkr} \quad (98)$$

Normalizing the field to the free space broadside field, the mode sum becomes

$$\left| \frac{E_\phi}{E_{\phi FS}} \right| = \left[ \frac{2\pi r^2}{k^2 a \sin(\frac{r}{a})} \right]^{1/2} \left| \sum_m \rho_m^{1/2} g_m(z) g_m(z_T) e^{-j\rho_m r} \right| \quad (99)$$

The coherent mode sum in decibels is given by

$$\begin{aligned} ECMS &= 10 \log \left| \frac{E_\phi}{E_{\phi FS}} \right|^2 \\ &= 10 \log \left\{ \frac{2\pi r^2}{k^2 a \sin(\frac{r}{a})} \left| \sum_m \rho_m^{1/2} g_m(z) g_m(z_T) e^{-j\rho_m r} \right|^2 \right\} \quad (100) \end{aligned}$$

The power sum or incoherent mode sum (EIMS) signal can also be calculated if the phase of each term is ignored:

$$EIMS = 10 \log \left\{ \frac{2\pi r^2}{k^2 a \sin(\frac{r}{a})} \sum_m \left| \rho_m^{1/2} g_m(z) g_m(z_T) e^{-j\rho_m r} \right|^2 \right\} \quad (101)$$

The path losses are

$$\text{Coherent path loss} = 32.45 + 20 \log r + 20 \log f - \text{ECMS} \quad (102)$$

and

$$\text{Incoherent path loss} = 32.45 + 20 \log r + 20 \log f - \text{EIMS} \quad (103)$$

where  $r$  is the range in kilometers and  $f$  is the frequency in MHz.

## B. VERTICALLY POLARIZED WAVE PROPAGATION

A vertically polarized wave may be treated as due to a radiating electric dipole oriented in the  $z$  direction and located at  $r=0$  and  $z = z_T$ .

The electric field and magnetic field can be approximated by [Ref. 1]

$$\vec{H} = j\epsilon_0\omega\vec{\nabla} \times \vec{\Pi} \quad (104)$$

and

$$\vec{E} = \vec{\nabla} \times \vec{\nabla} \times \vec{\Pi} \quad (105)$$

where  $\vec{\Pi}$  is the electric Hertz vector.

Calculation of the electric Hertz vector is identical to that of the magnetic Hertz vector except for the following modifications:

1. The height gain function satisfies the following differential equation:

$$\left\{ \frac{\partial^2}{\partial z^2} + k^2 \left[ m_i^2(z) - \frac{\rho^2}{k^2} \right] \right\} f_i(\rho, z) = -\frac{2\pi}{\epsilon_0} \delta(z - z_T) \quad (106)$$

2. Continuity condition at  $i=1$  becomes

$$n_g^2 f_1(\rho, 0) = n_1^2(0) f_1(\rho, 0) \quad (107)$$

This leads to

$$\Delta_{1,1} = \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 k'_1(q_{1,1}) - j\epsilon \frac{n_1^2(0)}{n_g^2} k_1(q_{1,1}) \quad (108)$$

$$\Delta_{1,2} = \left( \frac{k}{z_1} \right)^{2/3} z_1 k'_2(q_{1,1}) - j \frac{n_1^2(0)}{n_2^2} k_2(q_{1,1}) \quad (109)$$

and

$$\psi_1 = j \frac{n_1^2(0)}{n_2^2} f_1^{(p)}(\rho, q_{1,1}) \delta_{1M} - \frac{\hat{c}}{\hat{c}z} [f_1^{(p)}(\rho, q_{1,1})] \delta_{1M} \quad (110)$$

The transverse magnetic field component,  $H_\phi$ , is given by

$$H_\phi = -j\omega\epsilon_0 \frac{\partial \Pi}{\partial r} \quad (111)$$

From Equations 94 to 98, it can be seen that

$$\begin{aligned} \left| \frac{H_\phi}{H_{zFS}} \right| &= \left| \frac{E_\phi}{E_{zFS}} \right| \\ &= \left[ \frac{2\pi r^2}{k^2 a \sin(\frac{r}{a})} \right]^{1/2} \left| \sum_m \rho_m^{1/2} g_m(z) g_m(z_T) e^{-j\epsilon_m r} \right| \end{aligned} \quad (112)$$

It follows from Equation 112 that ECMS, EIMS, coherent path loss and incoherent path loss are given by Equations 100, 101, 102 and 103 respectively.

### C. DETERMINATION OF THE HEIGHT GAIN FUNCTION

The height gain function,  $f_i(\rho, z)$ , is given by

$$f_i(\rho, z) = B_i [A_i k_i(q_i) + k_2(q_i)] \quad ; \quad i = 1 \text{ to } I-1 \quad (113)$$

or

$$f_i(\rho, z) = B_i h_2(q_i) \quad ; \quad i = I \quad (114)$$

The coefficients,  $A_i$  and  $B_i$ , are functions of  $\rho$  as in Equation 25.

Determination of the values of  $A_i$  and  $B_i$  is dependent on the evaluation of the integral (or summation) of Equation 88. It was found that if the attenuation rate of a particular mode is small ( $\leq 0.1$  dB/km), the integration Equation 88 is carried out from the top layer to the first layer with the height gain function at the top layer set to a certain finite value. This will prevent the integral from blowing up.



On the other hand, if the rate of attenuation is large ( $> 0.1$  dB/km), the integration is performed from the first layer to the top layer with the height gain function at the first layer set to a certain finite value. This will ensure that the value of integral will not become too small.

Therefore, for attenuation rates greater than 0.1 dB/km,  $A_i$ 's and  $B_i$ 's are determined successively starting from  $A_1$  and  $B_1$  up to  $A_I$  and  $B_I$ . At  $i = 1$  and  $z = 0$ ,

$$f_0(\rho, z=0) = f_1(\rho, z=0) \quad (115)$$

and

$$\frac{\partial}{\partial z} [f_0(\rho, z)]_{z=0} = \frac{\partial}{\partial z} [f_1(\rho, z)]_{z=0} \quad (116)$$

Thus,

$$A_g = B_1 [A_1 k_1(q_{1,1}) + k_2(q_{1,1})] \quad (117)$$

and

$$j\omega A_g = B_1 [A_1 k'_1(q_{1,1}) + k'_2(q_{1,1})] \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 \quad (118)$$

Dividing Equation 118 by Equation 117 gives

$$\begin{aligned} A_1 &= \frac{\left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 k'_2(q_{1,1}) - j\omega k_2(q_{1,1})}{j\omega k_1(q_{1,1}) - \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 k'_1(q_{1,1})} \\ &= \frac{q'_1 k'_2(q_{1,1}) - j\omega k_2(q_{1,1})}{j\omega k_1(q_{1,1}) - q'_1 k'_1(q_{1,1})} \end{aligned} \quad (119)$$

where

$$q'_1 = \frac{dq_1}{dz} = \left( \frac{k}{\alpha_1} \right)^{2/3} \alpha_1 \quad (120)$$

In the program,  $f_1(\rho, z_1)$  is set to 1. This gives

$$B_1 = \frac{1}{A_1(\rho) k_1(q_{1,1}) + k_2(q_{1,1})} \quad (121)$$

For  $i=2$  to  $I-1$ ,  $A_i$  's and  $B_i$  's are determined by matching conditions at every layer boundary. At layer boundary  $z = z_i$ ,

$$f_i(\rho, z_i) = f_{i-1}(\rho, z_i) \quad (122)$$

and

$$\frac{\partial}{\partial z} [f_i(\rho, z_i)] = \frac{\partial}{\partial z} [f_{i-1}(\rho, z_i)] \quad (123)$$

From Equation 122,

$$\begin{aligned} B_i [A_i k_1(q_{i,i}) + k_2(q_{i,i})] \\ = B_{i-1} [A_{i-1} k_1(q_{i-1,i}) + k_2(q_{i-1,i})] \end{aligned} \quad (124)$$

Thus,

$$B_i = \frac{B_{i-1} [A_{i-1} k_1(q_{i-1,i}) + k_2(q_{i-1,i})]}{A_i k_1(q_{i,i}) + k_2(q_{i,i})} \quad (125)$$

The coefficient,  $A_i$ , is determined from Equation 123 as follows:

$$\begin{aligned} B_i [A_i k'_1(q_{i,i}) + k'_2(q_{i,i})] q'_i \\ = B_{i-1} [A_{i-1} k'_1(q_{i-1,i}) + k'_2(q_{i-1,i})] q'_{i-1} \end{aligned} \quad (126)$$

Substituting Equation 125 into Equation 126 gives

$$\begin{aligned} \frac{A_{i-1} k_1(q_{i-1,i}) + k_2(q_{i-1,i})}{A_i k_1(q_{i,i}) + k_2(q_{i,i})} [A_i k'_1(q_{i,i}) + k'_2(q_{i,i})] q'_i \\ = [A_{i-1} k'_1(q_{i-1,i}) + k'_2(q_{i-1,i})] q'_{i-1} \end{aligned} \quad (127)$$

Thus,

$$A_i \equiv \frac{n A_{i-1}}{d A_{i-1}} \quad (128)$$

where

$$\begin{aligned} nA_i &= q'_i k'_2(q_{i,i})[A_{i-1} k_1(q_{i-1,i}) + k_2(q_{i-1,i})] \\ &- q'_{i-1} k_2(q_{i,i})[A_{i-1} k'_1(q_{i-1,i}) + k'_2(q_{i-1,i})] \end{aligned} \quad (129)$$

and

$$\begin{aligned} dA_i &= q'_{i-1} k_1(q_{i,i})[A_{i-1} k'_1(q_{i-1,i}) + k'_2(q_{i-1,i})] \\ &- q'_i k'_1(q_{i,i})[A_{i-1} k_1(q_{i-1,i}) + k_2(q_{i-1,i})] \end{aligned} \quad (130)$$

For  $i = 1$  and at the layer boundary,  $z = z_l$ ,

$$f_l(\rho, z_l) = f_{l-1}(\rho, z_l) \quad (131)$$

This gives

$$B_l = \frac{B_{l-1}[A_{l-1} k_1(q_{l-1,l}) + k_2(q_{l-1,l})]}{h_2(q_{l,l})} \quad (132)$$

For attenuation rates less than or equal to 0.1 dB/km,  $A_i$ 's and  $B_i$ 's are determined successively starting from  $A_l$  and  $B_l$  down to  $A_1$  and  $B_1$ .

The height gain function at the top layer is given by

$$f_l(\rho, z_l) = h_2(q_l) \quad (133)$$

where the coefficient,  $B_l$ , is set to one. Matching the boundary condition at  $z = z_l$  gives

$$B_{l-1} = \frac{h_2(q_{l,l})}{A_{l-1} k_1(q_{l-1,l}) + k_2(q_{l-1,l})} \quad (134)$$

and

$$A_{l-1} = \frac{nA_{l-1}}{dA_{l-1}} \quad (135)$$

where

$$\begin{aligned} nA_{I-1} &= q'_{I-1} h'_2(q_{I,I}) k_2(q_{I-1,I}) \\ &- q'_{I-1} h_2(q_{I,I}) k'_2(q_{I-1,I}) \end{aligned} \quad (136)$$

and

$$\begin{aligned} dA_{I-1} &= q'_{I-1} h_2(q_{I,I}) k'_1(q_{I-1,I}) \\ &- q'_{I-1} h'_2(q_{I,I}) k_1(q_{I-1,I}) \end{aligned} \quad (137)$$

Similarly, the coefficients,  $A_i$  and  $B_i$ , are obtained by matching boundary conditions at  $z = z_i$  for  $i = I-2$  to 1.

Thus,

$$B_i = B_{i+1} \frac{A_{i+1} k_1(q_{i+1,i+1}) + k_2(q_{i+1,i+1})}{A_i k_1(q_{i,i+1}) + k_2(q_{i,i+1})} \quad (138)$$

and

$$A_i = \frac{nA_i}{dA_i} \quad (139)$$

where

$$\begin{aligned} nA_i &= q'_{i+1} k_2(q_{i,i+1}) [A_{i+1} k'_1(q_{i+1,i+1}) + k'_2(q_{i+1,i+1})] \\ &- q'_{i+1} k'_2(q_{i,i+1}) [A_{i+1} k_1(q_{i+1,i+1}) + k_2(q_{i+1,i+1})] \end{aligned} \quad (140)$$

and

$$\begin{aligned} dA_i &= q'_{i+1} k'_1(q_{i,i+1}) [A_{i+1} k_1(q_{i+1,i+1}) + k_2(q_{i+1,i+1})] \\ &- q'_{i+1} k_1(q_{i,i+1}) [A_{i+1} k'_1(q_{i+1,i+1}) + k'_2(q_{i+1,i+1})] \end{aligned} \quad (141)$$

#### D. TEST FOR EVANESCENCE

The functions,  $k_1(q)$  and  $k_2(q)$ , in Equation 26 and 27 can also be expressed in terms of the modified Hankel functions of order one third as follows (see Appendix A):

$$\begin{aligned}
k_1(q) &= -j 2(12)^{1/6} Ai(-q e^{j2\pi/3}) \\
&= \left( \frac{2}{3} q^{3/2} \right)^{1/3} H_{1/3}^{(1)} \left( \frac{2}{3} q^{3/2} \right) \\
&= h_1(q)
\end{aligned} \tag{142}$$

and

$$\begin{aligned}
k_2(q) &= 2(12)^{1/6} e^{j\pi/6} Ai(-q) \\
&= h_2(q) - e^{j4\pi/3} h_1(q) ,
\end{aligned} \tag{143}$$

where

$$h_1(q) \equiv \left( \frac{2}{3} q^{3/2} \right)^{1/3} H_{1/3}^{(1)} \left( \frac{2}{3} q^{3/2} \right) \tag{144}$$

and

$$\begin{aligned}
h_2(q) &= j 2(12)^{1/6} Ai(-q e^{-j2\pi/3}) \\
&\equiv \left( \frac{2}{3} q^{3/2} \right)^{1/3} H_{1/3}^{(2)} \left( \frac{2}{3} q^{3/2} \right) .
\end{aligned} \tag{145}$$

For large  $|q|$ ,  $k_1(q)$  and  $k_2(q)$  can be approximated by the asymptotic expansion of the Hankel function [Ref. 4 and Appendix A]. That is,

$$\begin{aligned}
k_1(q) &\sim (12)^{1/6} \pi^{-1/2} q^{-1/4} \exp \left[ j \left( \frac{2}{3} q^{3/2} - \frac{5\pi}{12} \right) \right] ; \\
-\frac{2\pi}{3} &< \arg(q) < \frac{4\pi}{3}
\end{aligned} \tag{146}$$

and

$$\begin{aligned}
k_2(q) &\sim (12)^{1/6} \pi^{-1/2} q^{-1/4} \exp \left[ -j \left( \frac{2}{3} q^{3/2} - \frac{5\pi}{12} \right) \right] ; \\
0 &< \arg(q) < 2\pi .
\end{aligned} \tag{147}$$

From Equations 146 and 147, it can be deduced that for large  $|q|$  and in the region

$$\frac{2\pi}{3} < \arg(q) < \frac{4\pi}{3}, \quad (148)$$

$k_1(q)$  will be exponentially large and  $k_2(q)$  exponentially small. When this occurs,  $h_1(q)$  and  $h_2(q)$  become linearly dependent numerically, i.e., from Equation 143.

$$h_2(q) \simeq e^{j4\pi/3} h_1(q) . \quad (149)$$

This implies that  $k_1(q)$  and  $k_2(q)$  are also linearly dependent numerically. Under this condition, the matrix of the modal function will become singular (in a numerical sense) for all values of  $q$ . Pappert and Goodhart defined the fields as evanescent when Equation 149 holds [Ref. 5 and 6].

When evanescence occurs, the height gain function is represented by an exponentially decaying field, i.e.,

$$f_i(\rho, z) = B_i k_2(q_i) . \quad (150)$$

It is necessary to incorporate test conditions in the program to check if the rf field has become evanescent. This is achieved by tracking the real part of the exponent of  $k_1(q)$  in Equation 146.

Since evanescence may only occur in the region specified by Equation 148, it can be deduced by simple transformation of Equation 148 that

$$\frac{3\pi}{2} < \arg \left[ \frac{2}{3} j q^{2/3} \right] < \frac{5\pi}{2} . \quad (151)$$

This shows that one needs only to test for evanescence when the real part of the exponent of  $k_1(q)$  is positive. That is,

$$\operatorname{Re} \left[ \frac{2}{3} j q^{2/3} \right] \geq 0 . \quad (152)$$

As the coefficients of the height gain function are dependent on how the integral of Equation 88 is evaluated, it follows that different test conditions exist for upward and downward integration. For upward integration, conditions for non-evanescence in the  $i^{\text{th}}$  layer are

$$\begin{aligned}
\operatorname{Re} \left[ \frac{2}{3} j (q_{i,i+1})^{3/2} \right] - \operatorname{Re} \left[ \frac{2}{3} j (q_{i,i})^{3/2} \right] &< 7 \\
\text{for } \operatorname{Re} \left[ \frac{2}{3} j (q_{i,i+1})^{3/2} \right] &\geq 0 \\
\text{and } \operatorname{Re} \left[ \frac{2}{3} j (q_{i,i})^{3/2} \right] &\geq 0
\end{aligned} \tag{153}$$

or

$$\begin{aligned}
\operatorname{Re} \left[ \frac{2}{3} j (q_{i,i+1})^{3/2} \right] &< 7 \quad \text{for } \operatorname{Re} \left[ \frac{2}{3} j (q_{i,i})^{3/2} \right] < 0 \\
\text{and } \operatorname{Re} \left[ \frac{2}{3} j (q_{i,i+1})^{3/2} \right] &\geq 0
\end{aligned} \tag{154}$$

or

$$\operatorname{Re} \left[ \frac{dq_i}{dz} \right] > 0 \tag{155}$$

or

$$z_{i+1} < 45 m \tag{156}$$

In principle, the right side of Equation 153 should be as large as possible. However, a large quantity will lead to severe cancellation error when extracting  $k_2(q)$  (which is a small number) from a linear combination of two large numbers,  $h_1(q)$  and  $h_2(q)$ .

Equation 154 is obtained from Equation 153 by setting  $\operatorname{Re} \left[ \frac{2}{3} j (q_{i,i})^{3/2} \right]$  to zero when its value is negative. This is true because when the real part of the exponent of  $k_1(q)$  is negative, the condition of evanescence will not occur.

With regards the condition in Equation 155, it can be shown that when  $\frac{d\eta_i}{dz}$  is small (which is usually the case) and Equation 155 is true, then

$$\operatorname{Re}(\alpha_i) > 0 \tag{157}$$

This implies

$$m^2(z_{i+1}) > m^2(z_i) \tag{158}$$

which leads to

$$\operatorname{Re}[q_{i,i+1}] > \operatorname{Re}[q_{i,i}] \quad (159)$$

or

$$\operatorname{Re} \left[ \frac{2}{3} j (q_{i,i+1})^{3/2} \right] < \operatorname{Re} \left[ \frac{2}{3} j (q_{i,i})^{3/2} \right] \quad (160)$$

or

$$\operatorname{Re} \left[ \frac{2}{3} j (q_{i,i+1})^{3/2} \right] - \operatorname{Re} \left[ \frac{2}{3} j (q_{i,i})^{3/2} \right] < 0 \quad (161)$$

By comparing Equation 161 with Equation 153, it is clear that Equation 161 is a condition for non-evanescence.

Finally, the condition in Equation 156 was based on the fact that at lower altitude, the magnitude of  $q$  is small enough such that  $h_1(q)$  and  $h_2(q)$  are well behaved in the region stated in Equation 148.

By the same argument, conditions for non-evanescence in the  $i^{\text{th}}$  layer for downward integration are as follows:

$$\begin{aligned} \operatorname{Re} \left[ \frac{2}{3} j (q_{i-1,i-1})^{3/2} \right] - \operatorname{Re} \left[ \frac{2}{3} j (q_{i-1,i})^{3/2} \right] &< 7 \\ \text{for } \operatorname{Re} \left[ \frac{2}{3} j (q_{i-1,i-1})^{3/2} \right] &\geq 0 \\ \text{and } \operatorname{Re} \left[ \frac{2}{3} j (q_{i-1,i})^{3/2} \right] &\geq 0 \end{aligned} \quad (162)$$

or

$$\begin{aligned} \operatorname{Re} \left[ \frac{2}{3} j (q_{i-1,i-1})^{3/2} \right] &< 7 \quad \text{for } \operatorname{Re} \left[ \frac{2}{3} j (q_{i-1,i})^{3/2} \right] < 0 \\ \text{and } \operatorname{Re} \left[ \frac{2}{3} j (q_{i-1,i-1})^{3/2} \right] &\geq 0 \end{aligned} \quad (163)$$

or

$$\operatorname{Re} \left[ \frac{dq_{i-1}}{dz} \right] < 0 \quad (164)$$



or

$$z_i < 45 m . \quad (165)$$

## E. EVALUATION OF THE MODAL FUNCTION

When evaluating the relative field strength of the electromagnetic waves using Equations 100 and 101, the eigenvalues,  $\rho = \rho_n$ , for which

$$\| \underline{\Delta} \| = 0 \quad (166)$$

are required. The determinant,  $\| \underline{\Delta} \|$ , is known as the modal function. Each eigenvalue represents a different electromagnetic mode that propagates in the multilayer tropospheric waveguide environment.

The roots of Equation 166 are found by the Shellman-Morlitt complex root-finding routine to be discussed in Chapter III. Instead of searching for solutions of Equation 166 in the  $\rho$ -space, it is more convenient to conduct the search in the  $q_{1,1}$ -plane. The variable,  $\rho$ , is related to  $q_{1,1}$  by

$$q_{1,1} = \left( \frac{k}{\alpha_1} \right)^{2/3} \left[ m^2(z_1) - \left( \frac{\rho_n}{k} \right)^2 \right] . \quad (167)$$

In searching for the roots of the modal function, the determinant must be evaluated many times. Therefore an efficient method for accomplishing this is required. In particular, the Laplace's expansion is used to obtain the determinant. The Laplace's expansion is based on the following theorem [Ref. 7].

Theorem: Laplace's Theorem states that if one selects any  $r$  rows of a determinant,  $|A|$ , forms all possible  $r$ -rowed minors from these  $r$  rows, multiplies each of these minors by its co-factor and then adds the results, one obtains  $|A|$ .

The method also takes advantage of the presence of zero elements in the determinant to reduce numerical computations.

The understanding of this technique can be best achieved with an example. Consider a trilinear refractivity profile with the following modal function:

$$\|\underline{\Delta}\|_3 = \begin{vmatrix} a_{11} & a_{12} & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 \\ a_{31} & a_{32} & a_{33} & a_{34} & 0 \\ 0 & 0 & a_{43} & a_{44} & a_{45} \\ 0 & 0 & a_{53} & a_{54} & a_{55} \end{vmatrix} . \quad (168)$$

Let the last two rows be the  $r$  rows from which minors are to be formed. Then applying Laplace's Theorem, the modal function is given by

$$\|\underline{\Delta}\|_3 = \begin{vmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{vmatrix} \cdot \begin{vmatrix} a_{44} & a_{45} \\ a_{54} & a_{55} \end{vmatrix} - \begin{vmatrix} a_{11} & a_{12} & 0 \\ a_{21} & a_{22} & a_{24} \\ a_{31} & a_{32} & a_{34} \end{vmatrix} \cdot \begin{vmatrix} a_{43} & a_{45} \\ a_{53} & a_{55} \end{vmatrix} , \quad (169)$$

If a four-layer piecewise linear refractivity profile is used, the modal function will be

$$\|\underline{\Delta}\|_4 = \begin{vmatrix} a_{11} & a_{12} & 0 & 0 & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 & 0 & 0 \\ a_{31} & a_{32} & a_{33} & a_{34} & 0 & 0 & 0 \\ 0 & 0 & a_{43} & a_{44} & a_{45} & a_{46} & 0 \\ 0 & 0 & a_{53} & a_{54} & a_{55} & a_{56} & 0 \\ 0 & 0 & 0 & 0 & a_{65} & a_{66} & a_{67} \\ 0 & 0 & 0 & 0 & a_{75} & a_{76} & a_{77} \end{vmatrix} . \quad (170)$$

Notice that Equation 170 is built from Equation 168 by adding eight non-zero entries, namely,  $a_{46}$ ,  $a_{56}$ ,  $a_{65}$ ,  $a_{66}$ ,  $a_{67}$ ,  $a_{75}$ ,  $a_{76}$  and  $a_{77}$ . Therefore, the results from Equation 168 can be used to compute Equation 170, i.e.,

$$\|\underline{\Delta}\|_4 = \|\underline{\Delta}\|_3 \cdot \begin{vmatrix} a_{66} & a_{67} \\ a_{76} & a_{77} \end{vmatrix} - \begin{vmatrix} a_{11} & a_{12} & 0 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & a_{24} & 0 \\ a_{31} & a_{32} & a_{33} & a_{34} & 0 \\ 0 & 0 & a_{43} & a_{44} & a_{45} \\ 0 & 0 & a_{53} & a_{54} & a_{55} \end{vmatrix} \cdot \begin{vmatrix} a_{65} & a_{67} \\ a_{75} & a_{77} \end{vmatrix} . \quad (171)$$

In Equation 171, the determinant of the  $5 \times 5$  matrix is also evaluated using the Laplace's expansion.

Without loss of generality, the evaluation of modal functions of multilayer piecewise linear refractivity profile is performed by systematically increasing the number of rows and columns of  $\underline{\Delta}$  and applying the Laplace's expansion.

The derivative of the modal function,  $\|\underline{\Delta}\|$ , with respect to  $q_{i,r}$  is also required by the Shellman-Morflitt root finding routine. This is accomplished by expanding  $\|\underline{\Delta}\|$  using the Laplace's expansion and performing differentiation using the chain-rule.

## F. EFFECTS OF SURFACE ROUGHNESS

The materials in this section are adapted from Chapter III of XWVG [Ref. 1]. It is included in this document for completeness.

Surface roughness could have significant effect on the signal levels at large ranges, especially when the frequencies of the rf waves are above several gigahertz.

The effects of surface roughness is modelled by Baumgartner in XWVG [Ref. 1] as a lossy infinitesimally thin bottom layer of constant modified refractivity shown in Figure 4.

In the bottom layer,  $0 \leq z \leq z_p$ , the height gain function,  $f_0(\rho, z)$ , can be written as

$$f_0(\rho, z) = A_0 [e^{j\mu z} + R e^{-j\mu z}] \quad (172)$$

where

$$\mu = k \sqrt{n_1^2(0) - \beta^2} \quad (173)$$

$$\beta^2 = \frac{\rho^2}{k^2} \quad (174)$$

and  $R$  is the plane wave reflection coefficient.

By matching boundary conditions at each layer boundary as was done in Section A, it can be seen that all previous results still hold if the following substitution is made:

$$\gamma \rightarrow \gamma_s = \mu \frac{1 - R}{1 + R} \quad (175)$$

In general, the bump heights of the rough surface are assumed to be Gaussian distributed. In this case, the reflection coefficient,  $R$ , can be represented as the product of the smooth surface plane wave reflection coefficient and a surface roughness factor [Ref. 8]. For horizontal polarization, the reflection coefficient,  $R$ , is given by

$$R = \begin{cases} R_H \exp \left[ -2k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right] & ; \quad \text{Re}(q_{1,1}) \geq 0 \\ R_H & ; \quad \text{Re}(q_{1,1}) < 0 \end{cases} \quad (176)$$

where  $\delta$  is the root mean square surface bump heights and  $R_H$  is the Fresnel reflection coefficient for a horizontally polarized wave given by

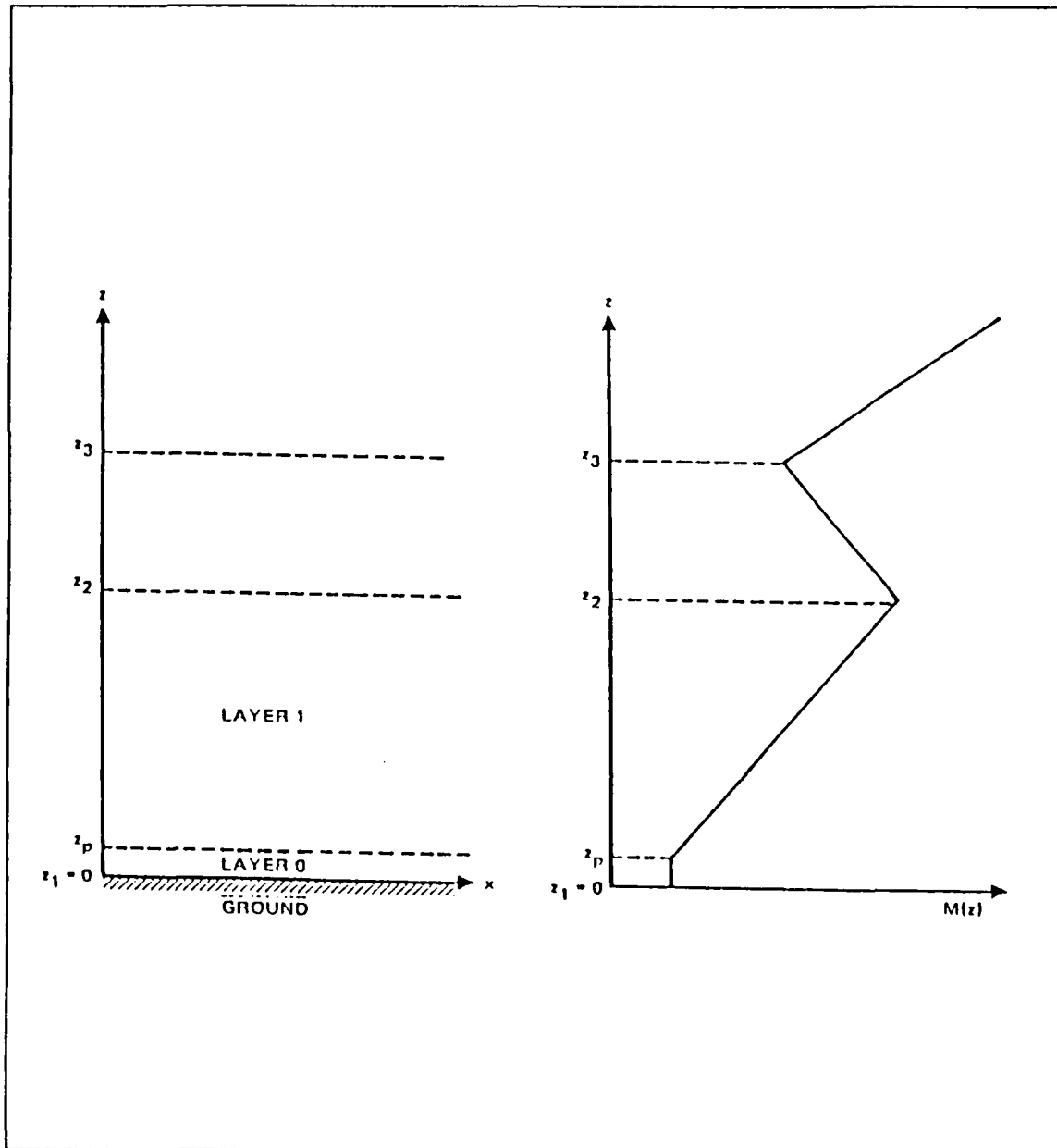


Figure 4. Effects of surface roughness modelled as layer 0 [From Ref. 1]

$$R_H = \frac{[n_1^2(0) - \beta^2]^{1/2} - [n_g^2 - \beta^2]^{1/2}}{[n_1^2(0) - \beta^2]^{1/2} + [n_g^2 - \beta^2]^{1/2}} \quad (177)$$

For vertical polarization, R is given by

$$R = \begin{cases} R_V \exp \left[ -2k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right] & ; \quad \text{Re}(q_{1,1}) \geq 0 \\ R_V & ; \quad \text{Re}(q_{1,1}) < 0 \end{cases}, \quad (178)$$

where  $R_V$  is the Fresnel reflection coefficient for a vertically polarized wave given by

$$R_V = \frac{n_g^2 [n_1^2(0) - \beta^2]^{1/2} - n_1^2(0) [n_g^2 - \beta^2]^{1/2}}{n_g^2 [n_1^2(0) - \beta^2]^{1/2} + n_1^2(0) [n_g^2 - \beta^2]^{1/2}}. \quad (179)$$

For horizontal polarization,  $r_H$  can be expressed as follows [Ref. 1]:

$$r_{SH} = k \frac{a(q_{1,1})}{b(q_{1,1})}, \quad (180)$$

where

$$a(q_{1,1}) = \left( \frac{\alpha_1}{k} \right)^{1/3} (q_{1,1})^{1/2} \tanh \left( \frac{\phi}{2} \right) + \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{1/2}, \quad (181)$$

$$b(q_{1,1}) = 1 + \left( \frac{k}{\alpha_1} \right)^{1/3} (q_{1,1})^{-1/2} \tanh \left( \frac{\phi}{2} \right) \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{1/2}, \quad (182)$$

$$T = n_g^2 - n^2(0) \quad (183)$$

and

$$\phi = 2k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1}. \quad (184)$$

For small values of  $|q_{1,1}|$ ,  $a(q_{1,1})$  and  $b(q_{1,1})$  can be expanded in series as

$$\begin{aligned}
a(q_{1,1}) = & \left( \frac{z_1}{k} \right)^{1/3} q_{1,1}^{-1/2} \left\{ k^2 \delta^2 \left( \frac{z_1}{k} \right)^{2/3} q_{1,1} \right. \\
& - \frac{1}{3} k^6 \delta^6 \left( \frac{z_1}{k} \right)^2 q_{1,1}^{-3} + \frac{2}{15} k^{10} \delta^{10} \left( \frac{z_1}{k} \right)^{10/3} q_{1,1}^{-5} - \dots \left. \right\} \\
& + T^{1/2} \left\{ 1 + \frac{1}{2T} \left( \frac{z_1}{k} \right)^{2/3} q_{1,1} - \frac{1}{8T^2} \left( \frac{z_1}{k} \right)^{4/3} q_{1,1}^{-2} \right. \\
& + \frac{1}{16T^3} \left( \frac{z_1}{k} \right)^2 q_{1,1}^{-3} - \frac{2}{128T^4} \left( \frac{z_1}{k} \right)^{8/3} q_{1,1}^{-4} \\
& \left. + \frac{7}{256T^5} \left( \frac{z_1}{k} \right)^{10/3} q_{1,1}^{-5} - \dots \right\} \quad (185)
\end{aligned}$$

and

$$\begin{aligned}
b(q_{1,1}) = & 1 + T^{1/2} q_{1,1}^{-1/2} \left\{ k^2 \delta^2 \left( \frac{z_1}{k} \right)^{1/3} + \frac{1}{2T} k^2 \delta^2 \left( \frac{z_1}{k} \right) q_{1,1} \right. \\
& - \left[ \frac{1}{3} k^6 \delta^6 + \frac{1}{8T^2} k^2 \delta^2 \right] \left( \frac{z_1}{k} \right)^{5/3} q_{1,1}^{-2} \\
& + \left[ -\frac{1}{6T} k^6 \delta^6 + \frac{1}{16T^3} k^2 \delta^2 \right] \left( \frac{z_1}{k} \right)^{7/3} q_{1,1}^{-3} \\
& + \left[ \frac{2}{15} k^{10} \delta^{10} + \frac{1}{24T^2} k^6 \delta^6 - \frac{5}{128T^4} k^2 \delta^2 \right] \left( \frac{z_1}{k} \right)^3 q_{1,1}^{-4} \\
& \left. + \left[ \frac{1}{15T} k^{10} \delta^{10} - \frac{1}{48T^3} k^6 \delta^6 + \frac{7}{256T^5} k^2 \delta^2 \right] \left( \frac{z_1}{k} \right)^{11/3} q_{1,1}^{-5} + \dots \right\} \quad (186)
\end{aligned}$$

The derivative of  $\gamma_{SH}$  with respect to  $q_{1,1}$  can be obtained by differentiating Equation 180, i.e.,

$$\frac{\hat{c}\gamma_{SH}}{\hat{c}q_{1,1}} = \frac{k}{b^2(q_{1,1})} \left[ b(q_{1,1}) \frac{\hat{c}a}{\hat{c}q_{1,1}} - a(q_{1,1}) \frac{\hat{c}b}{\hat{c}q_{1,1}} \right] \quad (187)$$

Similarly,  $\frac{\hat{c}a}{\hat{c}q_{1,1}}$  and  $\frac{\hat{c}b}{\hat{c}q_{1,1}}$  are obtained from Equation 181 and 182 respectively:

$$\begin{aligned}
\frac{\hat{c}a}{\hat{c}q_{1,1}} &= \frac{1}{2} q_{1,1}^{-1/2} \left( \frac{\alpha_1}{k} \right)^{1/3} \tanh \left( \frac{\phi}{2} \right) \\
&+ q_{1,1}^{1/2} \left( \frac{\alpha_1}{k} \right) k^2 \delta^2 \operatorname{sech}^2 \left( \frac{\phi}{2} \right) \\
&+ \frac{1}{2} \left( \frac{\alpha_1}{k} \right)^{2/3} \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{-1/2}
\end{aligned} \tag{188}$$

and

$$\begin{aligned}
\frac{\hat{c}b}{\hat{c}q_{1,1}} &= -\frac{1}{2} \left( \frac{k}{\alpha_1} \right)^{1/3} q_{1,1}^{3/2} \tanh \left( \frac{\phi}{2} \right) \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{1/2} \\
&+ \frac{1}{2} q_{1,1}^{-1/2} \left( \frac{\alpha_1}{k} \right)^{1/3} \tanh \left( \frac{\phi}{2} \right) \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{-1/2} \\
&+ q_{1,1}^{-1/2} k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{1/3} \operatorname{sech}^2 \left( \frac{\phi}{2} \right) \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{1/2} .
\end{aligned} \tag{189}$$

For small values of  $|q_{1,1}|$ ,  $\frac{\hat{c}a}{\hat{c}q_{1,1}}$  and  $\frac{\hat{c}b}{\hat{c}q_{1,1}}$  can be expanded into series as

$$\begin{aligned}
\frac{\hat{c}a}{\hat{c}q_{1,1}} &= \left( \frac{\alpha_1}{k} \right)^{1/3} q_{1,1}^{-1/2} \left\{ \frac{3}{2} k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{2/3} \right. \\
&- \frac{7}{6} k^6 \delta^6 \left( \frac{\alpha_1}{k} \right)^2 q_{1,1}^2 \\
&+ \left. \frac{11}{15} k^{10} \delta^{10} \left( \frac{\alpha_1}{k} \right)^{10/3} q_{1,1}^4 - \dots \right\} \\
&+ T^{1/2} \left\{ \frac{1}{2T} \left( \frac{\alpha_1}{k} \right)^{2/3} - \frac{1}{4T^2} \left( \frac{\alpha_1}{k} \right)^{4/3} q_{1,1} \right. \\
&+ \frac{3}{16T^3} \left( \frac{\alpha_1}{k} \right)^2 q_{1,1}^2 - \frac{5}{32T^4} \left( \frac{\alpha_1}{k} \right)^{8/3} q_{1,1}^3 \\
&+ \left. \frac{35}{256T^5} \left( \frac{\alpha_1}{k} \right)^{10/3} q_{1,1}^4 - \dots \right\}
\end{aligned} \tag{190}$$

and

$$\begin{aligned}
\frac{\partial \tilde{h}}{\partial q_{1,1}} = & T^{1/2} q_{1,1}^{-1/2} \left\{ \frac{1}{2} k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{1/3} + \frac{3}{4T} k^2 \delta^2 \left( \frac{\alpha_1}{k} \right) q_{1,1} \right. \\
& - \frac{5}{2} \left[ \frac{1}{3} k^6 \delta^6 + \frac{1}{8T^2} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{5/3} q_{1,1}^2 \\
& + \frac{7}{2} \left[ -\frac{1}{6T} k^6 \delta^6 + \frac{1}{16T^3} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{7/3} q_{1,1}^3 \\
& + \frac{9}{2} \left[ \frac{2}{15} k^{10} \delta^{10} + \frac{1}{24T^2} k^6 \delta^6 - \frac{5}{128T^4} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^3 q_{1,1}^4 \\
& \left. + \frac{11}{2} \left[ \frac{1}{15T} k^{10} \delta^{10} - \frac{1}{48T^3} k^6 \delta^6 + \frac{7}{256T^5} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{11/3} q_{1,1}^5 + \dots \right\} .
\end{aligned} \tag{191}$$

The corresponding results for vertical polarization are:

$$\gamma_{SV} = k \frac{\tilde{a}(q_{1,1})}{\tilde{b}(q_{1,1})} \tag{192}$$

and

$$\frac{\partial \gamma_{SV}}{\partial q_{1,1}} = \frac{k}{\tilde{b}^2(q_{1,1})} \left[ \tilde{b}(q_{1,1}) \frac{\partial \tilde{a}}{\partial q_{1,1}} - \tilde{a}(q_{1,1}) \frac{\partial \tilde{b}}{\partial q_{1,1}} \right] , \tag{193}$$

where

$$\tilde{a}(q_{1,1}) = \left( \frac{\alpha_1}{k} \right)^{1/3} q_{1,1}^{1/2} \tanh \left( \frac{\phi}{2} \right) + \frac{n_1^2(0)}{n_g^2} \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{1/2} \tag{194}$$

and

$$\tilde{b}(q_{1,1}) = 1 + \frac{n_1^2(0)}{n_g^2} q_{1,1}^{-1/2} \left( \frac{k}{\alpha_1} \right)^{1/3} \tanh \left( \frac{\phi}{2} \right) \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{1/2} . \tag{195}$$

For small values of  $|q_{1,1}|$ ,  $\tilde{a}(q_{1,1})$  and  $\tilde{b}(q_{1,1})$  becomes



$$\begin{aligned}
\tilde{a}(q_{1,1}) = & \left( \frac{\alpha_1}{k} \right)^{1/3} q_{1,1}^{1/2} \left\{ k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} - \frac{1}{3} k^6 \delta^6 \left( \frac{\alpha_1}{k} \right)^2 q_{1,1}^3 \right. \\
& + \frac{2}{15} k^{10} \delta^{10} \left( \frac{\alpha_1}{k} \right)^{10/3} q_{1,1}^5 - \dots \left. \right\} \\
& + \frac{n_1^2(0)}{n_g^2} T^{1/2} \left\{ 1 + \frac{1}{2T} \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right. \\
& - \frac{1}{8T^2} \left( \frac{\alpha_1}{k} \right)^{4/3} q_{1,1}^2 + \frac{1}{16T^3} \left( \frac{\alpha_1}{k} \right)^2 q_{1,1}^3 \\
& - \frac{5}{128T^4} \left( \frac{\alpha_1}{k} \right)^{8/3} q_{1,1}^4 + \frac{7}{256T^5} \left( \frac{\alpha_1}{k} \right)^{10/3} q_{1,1}^5 - \dots \left. \right\}
\end{aligned} \tag{196}$$

and

$$\begin{aligned}
\tilde{b}(q_{1,1}) = & 1 + \frac{n_1^2(0)}{n_g^2} T^{1/2} q_{1,1}^{1/2} \left\{ k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{1/3} + \frac{1}{2T} k^2 \delta^2 \left( \frac{\alpha_1}{k} \right) q_{1,1} \right. \\
& - \left[ \frac{1}{3} k^6 \delta^6 + \frac{1}{8T^2} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{5/3} q_{1,1}^2 \\
& + \left[ -\frac{1}{6T} k^6 \delta^6 + \frac{1}{16T^3} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{7/3} q_{1,1}^3 \\
& + \left[ \frac{2}{15} k^{10} \delta^{10} + \frac{1}{24T^2} k^6 \delta^6 - \frac{5}{128T^4} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^3 q_{1,1}^4 \\
& + \left[ \frac{1}{15T} k^{10} \delta^{10} - \frac{1}{48T^3} k^6 \delta^6 + \frac{7}{256T^5} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{11/3} q_{1,1}^5 + \dots \left. \right\}
\end{aligned} \tag{197}$$

The derivative of  $\tilde{a}(q_{1,1})$  and  $\tilde{b}(q_{1,1})$  are obtained from Equation 196 and 197 respectively:

$$\begin{aligned}
\frac{\partial \tilde{a}}{\partial q_{1,1}} = & \frac{1}{2} q_{1,1}^{-1/2} \left( \frac{\alpha_1}{k} \right)^{1/3} \tanh \left( \frac{\phi}{2} \right) \\
& + k^2 \delta^2 \frac{\alpha_1}{k} q_{1,1}^{1/2} \operatorname{sech}^2 \left( \frac{\phi}{2} \right) \\
& + \frac{1}{2} \frac{n_1^2(0)}{n_g^2} \left( \frac{\alpha_1}{k} \right)^{2/3} \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{-1/2}
\end{aligned} \tag{198}$$

and

$$\begin{aligned} \frac{\partial \tilde{h}}{\partial q_{1,1}} = & \frac{n_1^2(0)}{n_g^2} \left\{ -\frac{1}{2} q_{1,1}^{-3/2} \left( \frac{k}{\alpha_1} \right)^{1/3} \tanh \left( \frac{\phi}{2} \right) \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{1/2} \right. \\ & + \frac{1}{2} q_{1,1}^{-1/2} \left( \frac{\alpha_1}{k} \right)^{1/3} \tanh \left( \frac{\phi}{2} \right) \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{-1/2} \\ & \left. + k^2 \delta^2 q_{1,1}^{-1/2} \left( \frac{\alpha_1}{k} \right)^{1/3} \operatorname{sech}^2 \left( \frac{\phi}{2} \right) \left[ T + \left( \frac{\alpha_1}{k} \right)^{2/3} q_{1,1} \right]^{1/2} \right\} \quad (199) \end{aligned}$$

For small values of  $|q_{1,1}|$ ,  $\frac{\partial \tilde{a}}{\partial q_{1,1}}$  and  $\frac{\partial \tilde{h}}{\partial q_{1,1}}$  becomes

$$\begin{aligned} \frac{\partial \tilde{q}}{\partial q_{1,1}} = & \left( \frac{\alpha_1}{k} \right)^{1/3} q_{1,1}^{-1/2} \left\{ \frac{3}{2} k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{2/3} - \frac{7}{6} k^6 \delta^6 \left( \frac{\alpha_1}{k} \right)^2 q_{1,1}^{-2} \right. \\ & \left. + \frac{11}{15} k^{10} \delta^{10} \left( \frac{\alpha_1}{k} \right)^{10/3} q_{1,1}^{-4} - \dots \right\} \\ & + \frac{n_1^2(0)}{n_g^2} T^{1/2} \left\{ \frac{1}{2T} \left( \frac{\alpha_1}{k} \right)^{2/3} - \frac{1}{4T^2} \left( \frac{\alpha_1}{k} \right)^{4/3} q_{1,1} \right. \\ & + \frac{3}{16T^3} \left( \frac{\alpha_1}{k} \right)^2 q_{1,1}^{-2} - \frac{5}{32T^4} \left( \frac{\alpha_1}{k} \right)^{8/3} q_{1,1}^{-3} \\ & \left. + \frac{35}{256T^5} \left( \frac{\alpha_1}{k} \right)^{10/3} q_{1,1}^{-4} - \dots \right\} \quad (200) \end{aligned}$$

and

$$\begin{aligned}
\frac{\tilde{b}}{\tilde{c} q_{1,1}} = & \frac{n_1^2(0)}{n_g^2} T^{1/2} q_{1,1}^{-1/2} \left\{ \frac{1}{2} k^2 \delta^2 \left( \frac{\alpha_1}{k} \right)^{1/3} \right. \\
& + \frac{3}{4T} k^2 \delta^2 \left( \frac{\alpha_1}{k} \right) q_{1,1} - \frac{5}{2} \left[ \frac{1}{3} k^6 \delta^6 + \frac{1}{8T^2} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{5/3} q_{1,1}^2 \\
& + \frac{7}{2} \left[ -\frac{1}{6} k^6 \delta^6 + \frac{1}{16T^3} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{7/3} q_{1,1}^3 \\
& + \frac{9}{2} \left[ \frac{2}{15} k^{10} \delta^{10} + \frac{1}{24T^2} k^6 \delta^6 - \frac{5}{128T^4} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^3 q_{1,1}^4 \\
& \left. + \frac{11}{2} \left[ \frac{1}{15T} k^{10} \delta^{10} - \frac{1}{48T^3} k^6 \delta^6 + \frac{7}{256T^5} k^2 \delta^2 \right] \left( \frac{\alpha_1}{k} \right)^{11/3} q_{1,1}^5 + \dots \right\} .
\end{aligned} \tag{201}$$

## G. COMPLEX INDEX OF REFRACTION OF SEA WATER

The materials in this section are adapted from Chapter IV of XWVG [Ref. 1].

The complex index of refraction is one of the physical parameters required for evaluating the modal function and height gain functions. The complex index of refraction of sea water,  $n_g$ , is a function of frequency, temperature and salinity. It is related to the effective relative dielectric function,  $\epsilon_{eff}$ , and the effective conductivity,  $\sigma_{eff}$ , through

$$n_g^2 = \epsilon_{eff} - j \frac{\sigma_{eff}}{\omega \epsilon_0} , \tag{202}$$

where  $\epsilon_0$  is the permittivity of free space and  $\omega$  is the frequency at which  $n_g$  is evaluated. The effective relative dielectric function and the effective conductivity are related to the complex dielectric function,  $\epsilon$ , and the ionic conductivity,  $\sigma$ , through

$$\epsilon = \epsilon' - j \epsilon'' , \tag{203}$$

$$\epsilon_{eff} = \epsilon' \tag{204}$$

and

$$\sigma_{eff} = \sigma + \omega \epsilon_0 \epsilon'' . \tag{205}$$

In Equation 205,  $\sigma$  is assumed to be real. This is a good approximation for frequencies much less than 10<sup>6</sup> GHz [Ref. 1].

The theory of an ideal polar dielectric in an alternating electromagnetic field was first formulated by Debye [Ref. 9]. In the theory, the complex dielectric function is expressed as a function of a single relaxation time,  $\tau$ , as follows:

$$\epsilon = \epsilon' - j\epsilon'' \quad , \quad (206)$$

$$\epsilon' = \frac{\epsilon_s - \epsilon_\infty}{1 + \omega^2 \tau^2} \quad , \quad (207)$$

and

$$\epsilon'' = \frac{(\epsilon_s - \epsilon_\infty) \omega \tau}{1 + \omega^2 \tau^2} \quad . \quad (208)$$

where  $\epsilon_\infty$  is the part of the dielectric function due to the atomic and dielectric polarization,  $\epsilon_s$  is the static dielectric function and  $\omega$  is the frequency of the electromagnetic wave. Equations 206 to 208 represent the fall in the value of the dielectric function from  $\epsilon_s$  to  $\epsilon_\infty$ . This fall is accompanied by a single broad absorption band in the vicinity of the characteristic wavelength,  $\lambda_s$ , given by

$$\lambda_s = \frac{1}{\tau} \quad . \quad (209)$$

The characterisation of the variation with frequency of the complex dielectric function of water in terms of a single relaxation time is valid for frequencies up to 300 GHz [Ref. 1]. For higher frequencies, more than one relaxation time may be required.

In the model of Klein and Swift [Ref. 10],  $\epsilon_\infty$  is assumed to be 4.9 for both pure water and seawater. With this assumption, the static dielectric function for pure water is given by

$$\epsilon_{S,P} = 88.045 - 0.4147 t_c - 6.295 \times 10^{-4} t_c^2 + 1.075 \times 10^{-5} t_c^3 \quad (210)$$

and the static dielectric function of seawater is given by

$$\epsilon_{S,S} = \epsilon_{S,a} a(S, t_c) \quad , \quad (211)$$

where

$$\epsilon_{S,a} = 87.134 - 1.949 \times 10^{-1} t_c - 1.276 \times 10^{-2} t_c^2 + 2.491 \times 10^{-4} t_c^3 \quad (212)$$

and

$$\begin{aligned} a(S, t_c) = & 1.000 + 1.613 \times 10^{-5} S t_c - 3.656 \times 10^{-3} S \\ & + 3.210 \times 10^{-5} S^2 - 4.232 \times 10^{-7} S^3 . \end{aligned} \quad (213)$$

In the above equations,  $t_c$  is the temperature in degrees Celcius and  $S$  is the salinity in grams of salt per kilogram of seawater. The relaxation time of seawater is given by

$$\tau_S = \tau_p h(S, t_c) . \quad (214)$$

where  $\tau_p$  is the relaxation of pure water given by

$$\tau_p = 1.768 \times 10^{-11} - 6.086 \times 10^{-13} t_c + 1.104 \times 10^{-14} t_c^2 - 8.111 \times 10^{-17} t_c^3 \quad (215)$$

and

$$\begin{aligned} b(S, t_c) = & 1.000 + 2.282 \times 10^{-5} S t_c - 7.638 \times 10^{-4} S \\ & - 7.760 \times 10^{-6} S^2 + 1.105 \times 10^{-8} S^3 . \end{aligned} \quad (216)$$

The ionic conductivity of seawater in siemens per meter is given by

$$\sigma(t_c, S) = \sigma(25, S) \exp(-\Delta\varphi) . \quad (217)$$

where

$$\Delta = 25 - t_c , \quad (218)$$

$$\begin{aligned} \varphi = & 2.033 \times 10^{-2} + 1.266 \times 10^{-4} \Delta + 2.464 \times 10^{-6} \Delta^2 \\ & - S[1.849 \times 10^{-5} - 2.551 \times 10^{-7} \Delta + 2.551 \times 10^{-8} \Delta^2] \end{aligned} \quad (219)$$

and

$$\begin{aligned} \sigma(25, S) = & S[0.182521 - 1.46192 \times 10^{-3} S + 2.09324 \\ & \times 10^{-5} S^2 - 1.28205 \times 10^{-7} S^3] . \end{aligned} \quad (220)$$

## H. ATMOSPHERIC ABSORPTION

The materials in this section are adapted from References 11, 12 and 13.

The gas molecules in the air absorb energy from electromagnetic waves. The absorption is due to the quantum mechanical resonances of the gas molecules. These resonances at specific frequencies then broaden into bands as a result of collisions of the gas particles. In the lower atmosphere, the density of gas particles is higher. This means that collisions are more frequent and the absorption bands are broader.

In the troposphere (height < 10 km), only the lines bands of oxygen and water vapour are important. The principal resonance of oxygen is at 60 GHz and that of water vapour is at 22.2 GHz. The extent of absorption by oxygen molecules is dependent on the pressure,  $p$ , and temperature,  $t$ . In the case of water vapour, absorption is also dependent on the water vapour density,  $\rho$ . For specific values of  $p$ ,  $t$ , and  $\rho$ , the absorption coefficient,  $g_{at}$ , is computed as [Ref. 11]

$$g_{at} = 0.1820 f N_f \quad dB/km \quad , \quad (221)$$

where  $f$  is the frequency and  $N_f$  is the absorption spectrum. The absorption spectrum,  $N_f$ , is related to the line spectra of absorption,  $SF$ , the continuum spectra,  $N_c$ , and the liquid water extinction,  $N_w$ , by the following expressions [Ref. 11]:

$$N_f = \sum_i SF + N_w + N_c \quad ppm \quad . \quad (222)$$

The summation of  $SF$  is done over 44 spectral lines of oxygen and 29 spectral lines of water vapour, [Ref. 11], i.e.,

$$\begin{aligned} \sum_i SF = & \sum_{i=1}^{44} a_1(i) p \theta^3 \exp [a_2(i) (1 - \theta)] F \\ & + \sum_{i=1}^{29} b_1(i) c \theta^{3.5} \exp [b_2(i) (1 - \theta)] F \quad , \end{aligned} \quad (223)$$

where  $a_1(i)$ ,  $a_2(i)$ ,  $b_1(i)$  and  $b_2(i)$  are spectroscopic coefficients derived from experiments and spectroscopic parameter compilations. Values of  $a_1(i)$ ,  $a_2(i)$ ,  $b_1(i)$  and  $b_2(i)$  are listed in Table 1.

The dry air pressure,  $p$ , at a height of  $z$  above the earth's surface is given by [Ref. 12]

$$p = 101.325 \left[ 1 - \frac{0.0065z}{288} \right]^{5.256} \text{ kPa} \quad (224)$$

The relative inverse temperature,  $\theta$ , is related to the temperature,  $t_k$ , in  $^{\circ}\text{K}$  by

$$\theta = \frac{300}{t_k} \quad (225)$$

The water vapour partial pressure,  $e_w$ , in kPa is related to the saturation water vapour pressure,  $e_s$ , and relative humidity,  $rh$ , by the following equations:

$$e_w = e_s (rh) \times 10^{-2} \quad (226)$$

$$e_s = 0.6105 \exp \left[ 25.22 \left( \frac{t_c}{t_k} \right) - 5.31 \log \left( \frac{t_k}{273.15} \right) \right] \text{ kPa} \quad (227)$$

and

$$t_c = t_k - 273.15 \text{ } ^{\circ}\text{C} \quad (228)$$

In Equation 223,  $F$  is the shape factor of the broadened resonance lines given by [Ref. 11]

$$F = \left( \frac{f}{v_0} \right) \left[ \frac{g - (v_0 - f)\hat{c}}{(v_0 - f)^2 + g^2} + \frac{g - (v_0 + f)\hat{c}}{(v_0 + f)^2 + g^2} \right] \quad (229)$$

for oxygen line 1 to line 38 and

$$F = \left( \frac{f}{v_0} \right) \left[ \frac{g}{(v_0 - f)^2 + g^2} + \frac{g}{(v_0 + f)^2 + g^2} \right] \quad (230)$$

for oxygen line 39 to 44 and the water vapour lines.

In Equation 229 and 230,  $v_0$  is the molecular line centre frequency,  $g$  is the width and  $\hat{c}$  is the overlap interference.

For oxygen in air,

$$g = a_3 (p + 1.3 e) \theta^{0.9} \quad . \quad (231)$$

$$\hat{c} = a_4 p \theta^{0.5} \quad (232)$$

and for water vapour in air,

$$g = b_3 (4.8 e + p) \theta^{0.6} \quad . \quad (233)$$

The line data  $a_3$ ,  $a_4$  and  $b_3$  are listed in Table 1.

The continuum spectrum,  $N_c$ , is given by

$$N_c \simeq 1.9 f e p \theta^{3.1} \times 10^{-6} + 6.25 f p \theta^2 \times 10^{-4} \left[ \frac{g_0}{f^2 + g_0^2} + 2.1 p \theta^{0.5} \times 10^{-7} \right] \text{ ppm} \quad (234)$$

and

$$g_0 = 0.012(p + 1.3 e) \theta^{0.9} \quad . \quad (235)$$

The liquid water extinction,  $N_w$ , is evaluated from the following expressions:

$$N_w = \frac{4.49 \epsilon''}{(\epsilon' + 2)^2 + (\epsilon'')^2} \text{ ppm for } f \leq 300 \text{ GHz} \quad (236)$$

and

$$N_w \simeq 0.55 w f^{-0.1} \theta^{-6} \text{ ppm for } f > 300 \text{ GHz} \quad , \quad (237)$$

where  $w$  is the liquid water concentration in grams per cubic meter. The real and imaginary part of the dielectric function of water,  $\epsilon'$  and  $\epsilon''$ , are given by Equations 207 and 208 respectively. It is assumed that  $\epsilon_r$  and  $\tau$  given in Equations 207 and 208 are given by Equations 210 and 215 respectively.

The absorptivity of tropospheric gases,  $\eta$ , is related to the atmospheric absorption,  $g_{ab}$ , in dB km, by the following expression [Ref. 13]:

$$g_{ab} = \frac{2\pi}{\lambda} 10^6 \eta \log_{10} e \quad . \quad (238)$$



Hence,

$$\eta = \frac{g_{ab} \dot{\gamma}}{2\pi 10^{\tilde{a}} \log_{10} e} \quad (239)$$

Table 1. SPECTROSCOPIC COEFFICIENTS FOR OXYGEN ( $a_1 - a_6$ ) AND WATER ( $b_1 - b_3$ ) [FROM REF. 11]

Center frequency, $\nu_0$ (GHz)	Strength, $a_1$ (kHz/kPa)	Temperature exponent, $a_2$	Width, $a_3$ (GHz/kPa)	Interference, $a_4$ (1/kPa)	Temperature exponent, $a_5$	Quantum number identification, $K^*$
0°	0.307 E-3 (ppm/kPa)		0.12 (0.056) E-1			1° to 37°
50.47381	0.94 E-6	0.969 E+1	0.86 E-2	0.520 E-2	0.179 E+1	37°
50.98742	0.244 E-5	0.869 E+1	0.87 E-2	0.550 E-2	0.169 E+1	35°
51.50311	0.604 E-5	0.774 E+1	0.89 E-2	0.560 E-2	0.177 E+1	33°
52.02124	0.141 E-4	0.684 E+1	0.92 E-2	0.550 E-2	0.181 E+1	31°
52.54227	0.308 E-4	0.600 E+1	0.94 E-2	0.569 E-2	0.179 E+1	29°
53.06683	0.637 E-4	0.522 E+1	0.97 E-2	0.528 E-2	0.189 E+1	27°
53.59570	0.124 E-3	0.448 E+1	0.100 E-1	0.544 E-2	0.183 E+1	25°
54.12997	0.2265 E-3	0.381 E+1	0.102 E-1	0.480 E-2	0.199 E+1	23°
54.67115	0.3893 E-3	0.319 E+1	0.105 E-1	0.484 E-2	0.190 E+1	21°
55.22137	0.6274 E-3	0.262 E+1	0.1079 E-1	0.417 E-2	0.207 E+1	19°
55.78382	0.9471 E-3	0.212 E+1	0.1110 E-1	0.375 E-2	0.207 E+1	17°
56.26477	0.5453 E-3	0.100 E-1	0.1646 E-1	0.774 E-2	0.890	D1° 1°
56.36339	0.1335 E-2	0.166 E+1	0.1144 E-1	0.297 E-2	0.229 E+1	15°
56.96818	0.1752 E-2	0.126 E+1	0.1181 E-1	0.212 E-2	0.253 E+1	13°
57.61249	0.2125 E-2	0.910	0.1221 E-1	0.940 E-3	0.376 E+1	11°
58.32389	0.2369 E-2	0.621	0.1266 E-1	-0.550 E-3	-0.111 E+2	D2 9°
58.44658	0.1447 E-2	0.827 E-1	0.1449 E-1	0.597 E-2	0.790	3°
59.16422	0.2387 E-2	0.386	0.1319 E-1	-0.244 E-2	0.700 E-1	7°
59.59098	0.2097 E-2	0.207	0.1360 E-1	0.344 E-2	0.490	5°
60.30604	0.2109 E-2	0.207	0.1382 E-1	-0.435 E-2°	0.680	D3 5°
60.43478	0.2444 E-2	0.386	0.1297 E-1	0.132 E-2	-0.120 E+1	7°
61.15057	0.2486 E-2	0.621	0.1248 E-1	-0.360 E-3	0.584 E+1	9°
61.80016	0.2281 E-2	0.910	0.1207 E-1	-0.159 E-2	0.286 E+1	11°
62.41122	0.1919 E-2	0.126 E+1	0.1171 E-1	-0.266 E-2	0.226 E+1	D4 13°
62.48626	0.1507 E-2	0.827 E-1	0.1468 E-1	-0.503 E-2°	0.850	3°
62.99797	0.1492 E-2	0.166 E+1	0.1139 E-1	-0.334 E-2	0.218 E+1	15°
63.56852	0.1079 E-2	0.212 E+1	0.1108 E-1	-0.417 E-2	0.196 E+1	17°
64.12778	0.7281 E-3	0.262 E+1	0.1078 E-1	-0.448 E-2	0.200 E+1	19°
64.67886	0.4601 E-3	0.319 E+1	0.105 E-1	-0.515 E-2	0.184 E+1	21°
65.22412	0.2727 E-3	0.381 E+1	0.102 E-1	-0.507 E-2	0.192 E+1	23°
65.76474	0.152 E-3	0.448 E+1	0.100 E-1	-0.567 E-2	0.178 E+1	25°
66.30195	0.794 E-4	0.522 E+1	0.97 E-2	-0.549 E-2	0.184 E+1	27°
66.83663	0.391 E-4	0.600 E+1	0.94 E-2	-0.588 E-2	0.174 E+1	29°
67.36933	0.181 E-4	0.684 E+1	0.93 E-2	-0.560 E-2	0.177 E+1	31°
67.90051	0.795 E-5	0.774 E+1	0.89 E-2	-0.580 E-2	0.173 E+1	33°
68.43054	0.328 E-5	0.869 E+1	0.87 E-2	-0.570 E-2	0.165 E+1	35°
68.95972	0.128 E-5	0.969 E+1	0.86 E-2	-0.530 E-2	0.174 E+1	37°
118.75034	0.9341 E-3	0.000	0.1592 E-1	-0.441 E-3	0.890	1°

Table 1. (continued)

Center frequency, $\nu_0$ (GHz)	Strength, $a_1$ (kHz/kPa)	Temperature exponent, $a_2$	Width, $a_3$ (GHz/kPa)	Quantum number identification ( $O_2$ )	
				Lower	Upper
368.498350	0.679 E-4	0.200 E-1	0.156 E-1	1, 1	2, 3
424.763120	0.638 E-3	0.122 E-1	0.147 E-1	2, 1	2, 3
487.249371	0.235 E-3	0.122 E-1	0.147 E-1	2, 1	3, 3
715.393150	0.996 E-4	0.891 E-1	0.144 E-1	3, 3	4, 5
773.839732	0.571 E-3	0.798 E-1	0.140 E-1	4, 3	4, 5
834.145790	0.180 E-3	0.798 E-1	0.140 E-1	4, 3	5, 5
Identification ( $H_2O$ )					
$\nu_0$	$b_1$	$b_2$	$b_3$	Lower	Upper
• 22.235080	0.105	0.214 E+1	0.281 E-1	5, 2, 3	6, 1, 6
68.052	0.180 E-2	0.875 E+1	0.280 E-1	3, 2, 1 (1) <sup>a</sup>	4, 1, 3
• 183.310091	0.238 E+1	0.653	0.282 E-1	2, 2, 0	3, 1, 3
321.225644	0.460 E-1	0.616 E+1	0.220 E-1	9, 3, 6	10, 2, 9
• 325.152919	0.155 E+1	0.152 E+1	0.290 E-1	4, 2, 2	5, 1, 5
• 380.197372	0.123 E+2	0.102 E+1	0.285 E-1	3, 2, 1	4, 1, 4
386.778	0.400 E-2	0.733 E+1	0.160 E-1	11, 2, 10	10, 3, 7
437.34667	0.630 E-1	0.502 E+1	0.150 E-1	6, 6, 0	7, 5, 3
439.150812	0.921	0.356 E+1	0.175 E-1	5, 5, 0	6, 4, 3
443.018295	0.191	0.502 E+1	0.148 E-1	6, 6, 1	7, 5, 2
• 448.001075	0.107 E+2	0.137 E+1	0.246 E-1	3, 3, 0	4, 2, 3
470.888947	0.328	0.357 E+1	0.181 E-1	5, 5, 1	6, 4, 2
474.689127	0.124 E+1	0.234 E+1	0.210 E-1	4, 4, 0	5, 3, 3
488.491133	0.256	0.281 E+1	0.222 E-1	7, 1, 7	6, 2, 4
504.219	0.380 E-1	0.669 E+1	0.127 E-1	7, 7, 0	8, 6, 3
505.126	0.120 E-1	0.669 E+1	0.130 E-1	7, 7, 1	8, 6, 2
• 556.936002	0.526 E+3	0.114	0.317 E-1	1, 0, 1	1, 1, 0
• 620.700807	0.521 E+1	0.234 E+1	0.216 E-1	4, 4, 1	5, 3, 2
658.340	0.460	0.776 E+1	0.328 E-1	1, 0, 1 (1)	1, 1, 0
• 752.033227	0.259 E+3	0.336	0.302 E-1	2, 0, 2	2, 1, 1
836.836	0.120 E-1	0.811 E+1	0.170 E-1	11, 2, 9	10, 5, 6
859.810	0.150 E-1	0.799 E+1	0.270 E-1	2, 0, 2 (1)	2, 1, 1
899.380	0.910 E-1	0.784 E+1	0.300 E-1	1, 1, 1 (1)	2, 0, 2
903.280	0.640 E-1	0.835 E+1	0.280 E-1	2, 2, 1 (1)	3, 1, 2
907.773	0.179	0.504 E+1	0.204 E-1	8, 3, 5	9, 2, 8
• 916.169	0.890 E+1	0.137 E+1	0.249 E-1	3, 3, 1	4, 2, 2
970.320	0.940 E+1	0.184 E+1	0.246 E-1	4, 3, 1	5, 2, 4
• 987.910	0.145 E+3	0.180	0.299 E-1	1, 1, 1	2, 0, 2
• 1097.368	0.840 E+3	0.656	0.335 E-1	3, 0, 3	3, 1, 2

Read 0.307 E-3 as  $0.307 \times 10^{-3}$ .<sup>a</sup>Nonresonant  $O_2$  spectrum (equation (21)).<sup>b</sup>D1 denotes doublet.<sup>c</sup>Rosenkranz's [1975] first-order solution for the 60-GHz band shape has a shortcoming: the listed values of  $a_2$  ( $K = 3^-, 5^-$ ) have to be reduced 5% (which is of negligible consequence) to assure that  $\alpha(O_2) \geq 0.000$  for  $f > 160$  GHz even when  $N_2^+ = 0$  (equation (21)).<sup>d</sup>Here (1) denotes first vibrationally excited state.

### III. SHELLMAN - MORFITT ROOT FINDING ROUTINE

#### A. GENERAL OUTLINE

The Shellman-Morfitt root finding routine was developed by C.H. Shellman and D.G. Morfitt to locate ELF VLF LF mode constants of wave propagating in an earth-ionosphere waveguide environment. The theory of the algorithm described in this chapter is based on References 1 and 14.

The complex root finding routine is used to find the zeros of the modal function. The routine will locate all simple zeros of an analytic function in a prescribed rectangular region of the complex z-plane. The principle of the root finding method is based on the following theorem [Ref. 15]:

Theorem (Argument Principle):

Let D be a simply connected domain and  $f(z)$  be analytic in D except at a finite number of poles. Let  $\Gamma$  be a closed contour in D not passing through any of the zeros or poles of  $f(z)$ . Then, the accumulated phase change,  $\Delta \phi$ , of  $f(z)$  around  $\Gamma$  traversed in a clockwise direction is given by

$$\Delta \phi = 2\pi (N_z - N_p) \quad . \quad (240)$$

where  $N_z$  is the number of zeros and  $N_p$  is the number of poles (counting multiplicities) enclosed by  $\Gamma$ .

From the above theorem, if  $f(z)$  has only zeros in the region enclosed by  $\Gamma$ , then every constant phase contour associated with a zero crosses  $\Gamma$ .

An analytic function  $f(z)$  can be expressed as

$$f(z) = \sqrt{[Re(f)]^2 + [Im(f)]^2} \exp(j\theta) \quad , \quad (241)$$

where  $\theta$  is the phase of  $f(z)$  given by

$$\theta = \arg [f(z)] \quad . \quad (242)$$

If  $f(z)$  does not have any pole or branch point, then the constant phase curve,  $\theta = \theta_c$ , radiating from a zero of  $f(z)$  must cross the closed contour,  $\Gamma$ , enclosing the zero at least once. In addition, no other zero of  $f(z)$  may be on this phase curve. This is illustrated in Figure 5.

A line of constant phase which crosses the contour  $\Gamma$  may be followed until it leads to a zero or until it crosses the contour again. A zero of  $f(z)$  can be determined from the intersection of the curves

$$\operatorname{Im}(f) = 0 \quad ; \quad (\theta_c = 0 \text{ or } \pi) \quad (243)$$

and

$$\operatorname{Re}(f) = 0 \quad ; \quad \left( \theta_c = \frac{\pi}{2} \text{ or } \frac{3\pi}{2} \right) . \quad (244)$$

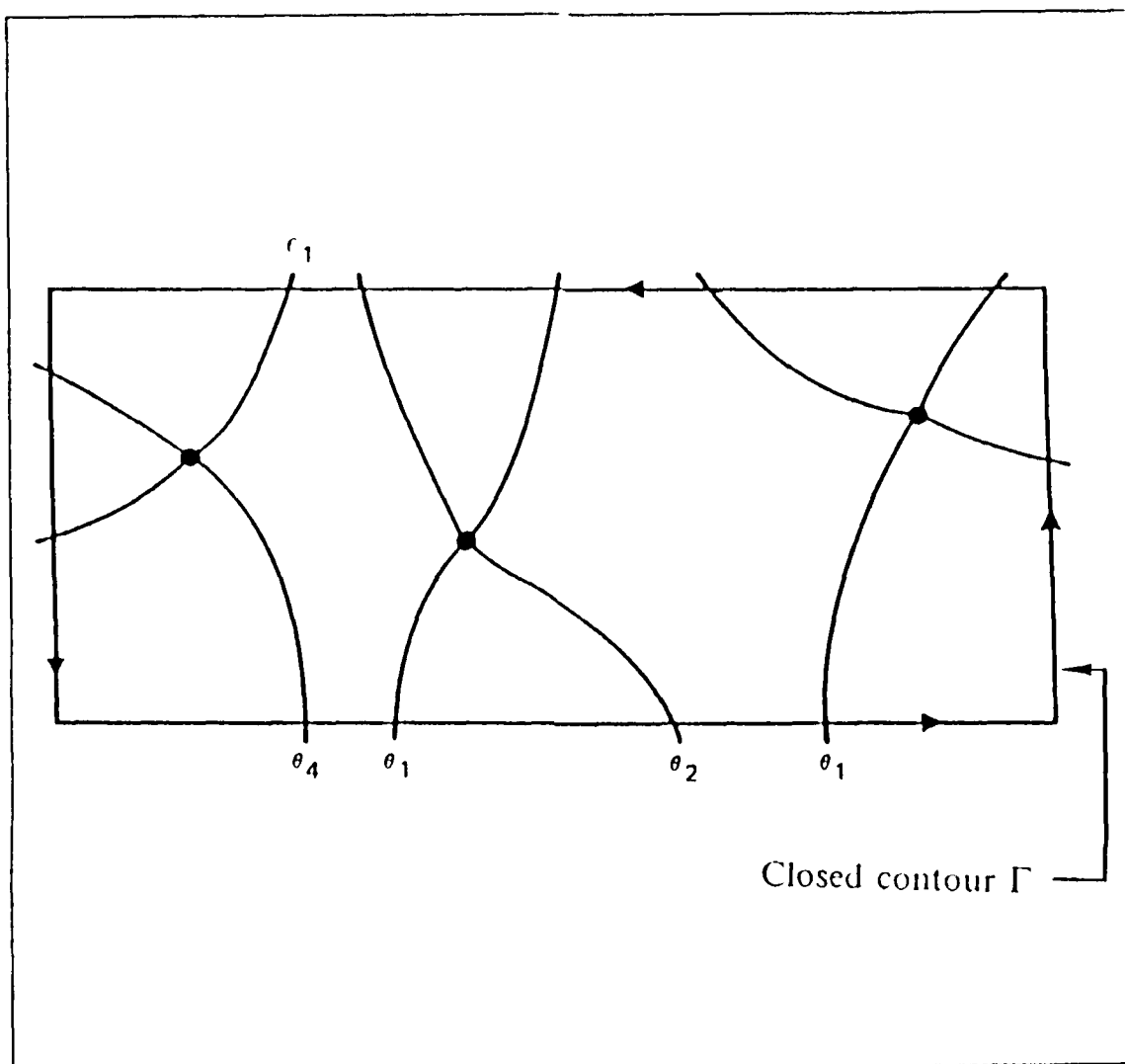


Figure 5. Constant Phase Line of  $f(z)$  [From Ref. 1]

## B. ROOT SEARCH PROCEDURE

A search rectangle is set up in the complex  $z$ -plane so that the simple zeros of  $f(z)$  will be located by the routine. The corners of this rectangle are denoted by:

- $t_L$  - value of the real part of  $z$  at the left edge of the rectangle.
- $t_R$  - value of the real part of  $z$  at the right edge of the rectangle.
- $t_T$  - value of the imaginary part of  $z$  at the top edge of the rectangle.
- $t_B$  - value of the imaginary part of  $z$  at the bottom of the rectangle.

The search rectangle is divided into small mesh squares of length  $\delta$ . Restrictions on the size of  $\delta$  will be addressed later.

A new search rectangle is generated from the specified search rectangle by expressing the edge of the new rectangle in terms of mesh units. One mesh unit is of length  $\delta$ . Thus

$$J_L = \text{Int} \left( \frac{t_L}{\delta} \right) \quad , \quad (245)$$

$$J_R = \text{Int} \left( \frac{t_R}{\delta} \right) \quad , \quad (246)$$

$$J_T = \text{Int} \left( \frac{t_T}{\delta} \right) \quad (247)$$

and

$$J_B = \text{Int} \left( \frac{t_B}{\delta} \right) \quad , \quad (248)$$

where  $\text{Int}(x)$  denotes the integer part of  $x$ . In order that the original search rectangle falls completely within the new search rectangle, the new rectangle is made one mesh unit larger on all sides. That is,

$$J_L = \begin{cases} J_L - 1 & : t_L \geq 0 \\ J_L - 2 & : t_L < 0 \end{cases} \quad , \quad (249)$$

$$J_R = \begin{cases} J_R + 2 & : t_R \geq 0 \\ J_R + 1 & : t_R < 0 \end{cases} \quad , \quad (250)$$

$$J_T = \begin{cases} J_T + 2 & : \quad t_T > 0 \\ J_T + 1 & : \quad t_T \leq 0 \end{cases} \quad (251)$$

and

$$J_B = \begin{cases} J_B - 1 & ; \quad t_B \geq 0 \\ J_B - 2 & ; \quad t_B < 0 \end{cases} \quad (252)$$

The search rectangle with mesh grid is shown in Figure 6. The plus and minus signs at the corners of the mesh squares are the signs of  $\text{Im}(f)$ . The solid curves represent the phase line,  $\text{Im}(f)=0$ , while the dashed curves represent the phase line,  $\text{Re}(f)=0$ . The zeros of  $f(z)$  are located at the crossing of the  $\text{Im}(f)=0$  and  $\text{Re}(f)=0$  curves. A local mesh coordinate system is shown in Figure 7. The lower left-hand corner of the mesh square is taken as the origin of that mesh square's local coordinate system. The value of  $f(z)$  evaluated in the local mesh coordinates of the  $k^{\text{th}}$  mesh square is denoted by  $f^{(k)}(\chi_k, \omega_k)$ . Thus, the values of  $f(z)$  at the corners of mesh square 1 are

$$f^{(1)}(0, 1) = f(J_L \delta + j J_T \delta) \quad , \quad (253)$$

$$f^{(1)}(0, 0) = f[J_L \delta + j (J_T - 1) \delta] \quad , \quad (254)$$

$$f^{(1)}(1, 0) = f[(J_L + 1) \delta + j (J_T - 1) \delta] \quad (255)$$

and

$$f^{(1)}(1, 1) = f[(J_L + 1) \delta + j J_T \delta] \quad . \quad (256)$$

A basic assumption made on  $f(z)$  is that  $\text{Im}(f)$  and  $\text{Re}(f)$  are linear functions of the local mesh coordinates,  $\chi_k$  and  $\omega_k$ . This means that in the mesh square  $k$ ,  $f(z)$  can be expressed as [Ref. 1]

$$f^{(k)}(\chi_k, \omega_k) = a_k + b_k \omega_k + c_k \chi_k + d_k \omega_k \chi_k \quad , \quad (257)$$



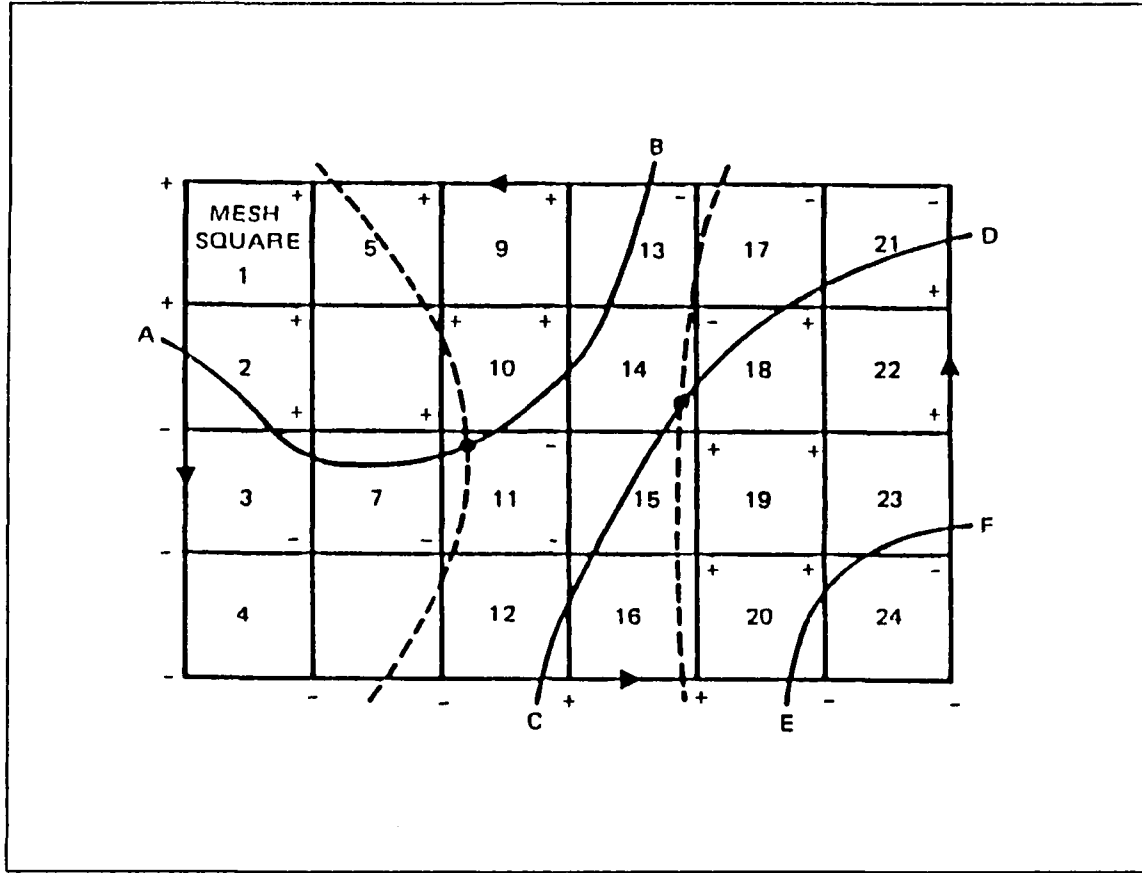


Figure 6. Illustration of search rectangle with mesh grid [From Ref. 1]

where

$$a_k = f^{(k)}(0, 0) \quad , \quad (258)$$

$$b_k = f^{(k)}(0, 1) - f^{(k)}(0, 0) \quad , \quad (259)$$

$$c_k = f^{(k)}(1, 0) - f^{(k)}(0, 0) \quad (260)$$

and

$$d_k = f^{(k)}(0, 0) + f^{(k)}(1, 1) - f^{(k)}(0, 1) - f^{(k)}(1, 0) \quad . \quad (261)$$

This assumption puts an upper bound on the size of the mesh unit,  $\delta$ . The mesh unit should be sufficiently small to ensure that the linear variation of  $\hat{f}(z)$  holds. However,  $\delta$  should not be excessively small to reduce computer run time.

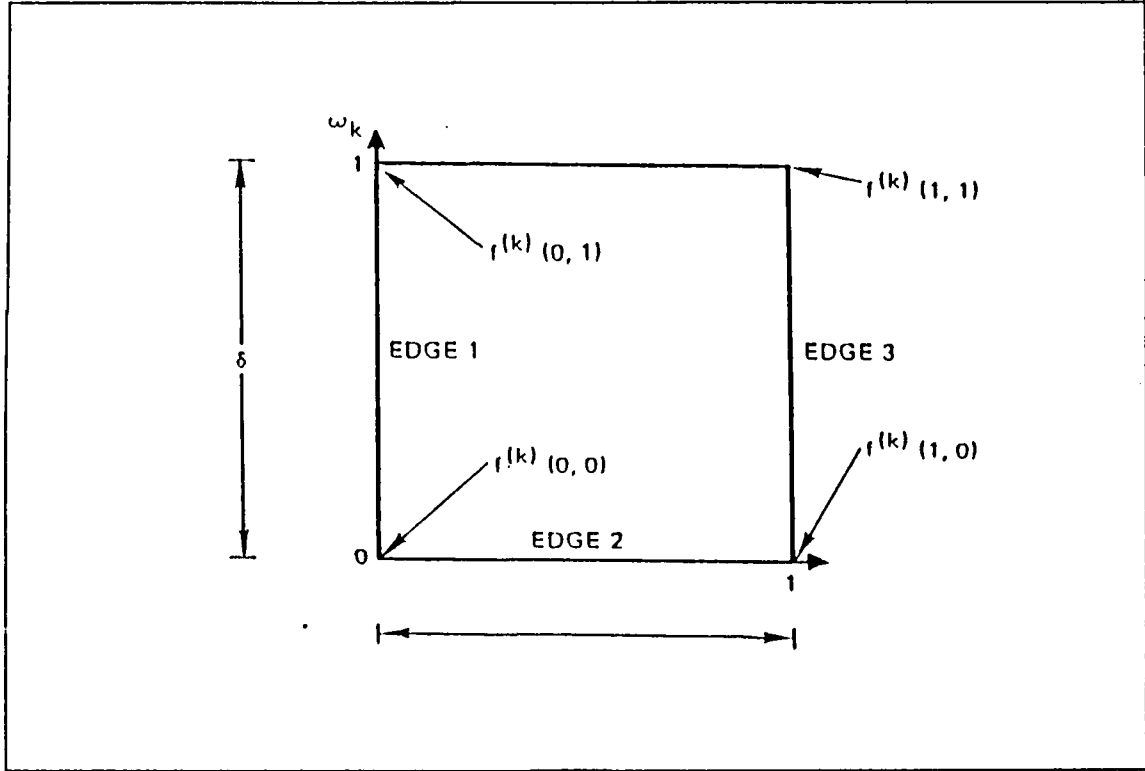


Figure 7. Local mesh coordinate system [From Ref. 1]

The zeros of  $f(z)$  are searched by examining the imaginary part of  $f(z)$  counter-clockwise around the rectangular contour. For the example shown in Figure 6, the edges of the search rectangle are examined for crossing of the phase curve,  $\text{Im}(f)=0$ , starting with mesh square 1, edge 1. The values  $\text{Im}[f^{(k)}(0,1)]$  and  $\text{Im}[f^{(k)}(0,0)]$  are first computed. If they are of the same sign, then there will be no  $\text{Im}(f)=0$  curve passing through edge 1.

Mesh square 2 is investigated next. The values of  $\text{Im}[f^{(k)}(0,1)]$  and  $\text{Im}[f^{(k)}(0,0)]$  are computed. The value of  $\text{Im}[f^{(k)}(0,1)]$  is identical to  $\text{Im}[f^{(k)}(0,0)]$ . If the signs of  $\text{Im}[f^{(k)}(0,1)]$  and  $\text{Im}[f^{(k)}(0,0)]$  are opposite as shown in Figure 6, then the  $\text{Im}(f)=0$  curve enters mesh square 2 along edge 1. The crossing point is given by [Ref. 1]

$$\chi_2 = 0 \quad (262)$$

and

$$\omega_2 = - \frac{Im[f^{(2)}(0, 0)]}{Im[f^{(2)}(0, 1)] - Im[f^{(2)}(0, 0)]} \quad (263)$$

The next step is to determine if a zero is within mesh square 2. The values of  $Im[f^{(2)}(1, 0)]$  and  $Im[f^{(2)}(1, 1)]$  are first computed. Two tests must be made:

1. A test is made to determine if there is one or two  $Im(f)=0$  curves entering and leaving the mesh square. If  $Im[f^{(2)}(0, 0)]$  and  $Im[f^{(2)}(1, 1)]$  are of the same sign, e.g., minus, and  $Im[f^{(2)}(0, 1)]$  and  $Im[f^{(2)}(1, 0)]$  both have opposite signs, e.g., plus, then there are two lines of  $Im(f)=0$  entering and leaving the square. This condition is depicted in mesh square 14 of Figure 6. Otherwise, there is only one line of  $Im(f)=0$  entering and leaving the mesh square. This is illustrated in mesh square 2 of Figure 6.
2. A test has to be performed to determine if there is at least one  $Re(f)=0$  entering and leaving the mesh square. If  $Re[f^{(2)}(0, 0)]$ ,  $Re[f^{(2)}(0, 1)]$ ,  $Re[f^{(2)}(1, 0)]$  and  $Re[f^{(2)}(1, 1)]$  are all of the same sign, then there is no line of  $Re(f)=0$  passing through the mesh square. Otherwise, there is at least one  $Re(f)=0$  curve entering and leaving the mesh square.

Note that the above tests will fail if there are two or more  $Im(f)=0$  or  $Re(f)=0$  crossing the mesh square at the same edge. As shown in Figure 8, two  $Im(f)=0$  crossing the same edge of a mesh square results in a zero being missed. In this situation, the mesh size has to be reduced in order to resolve the two zeros.

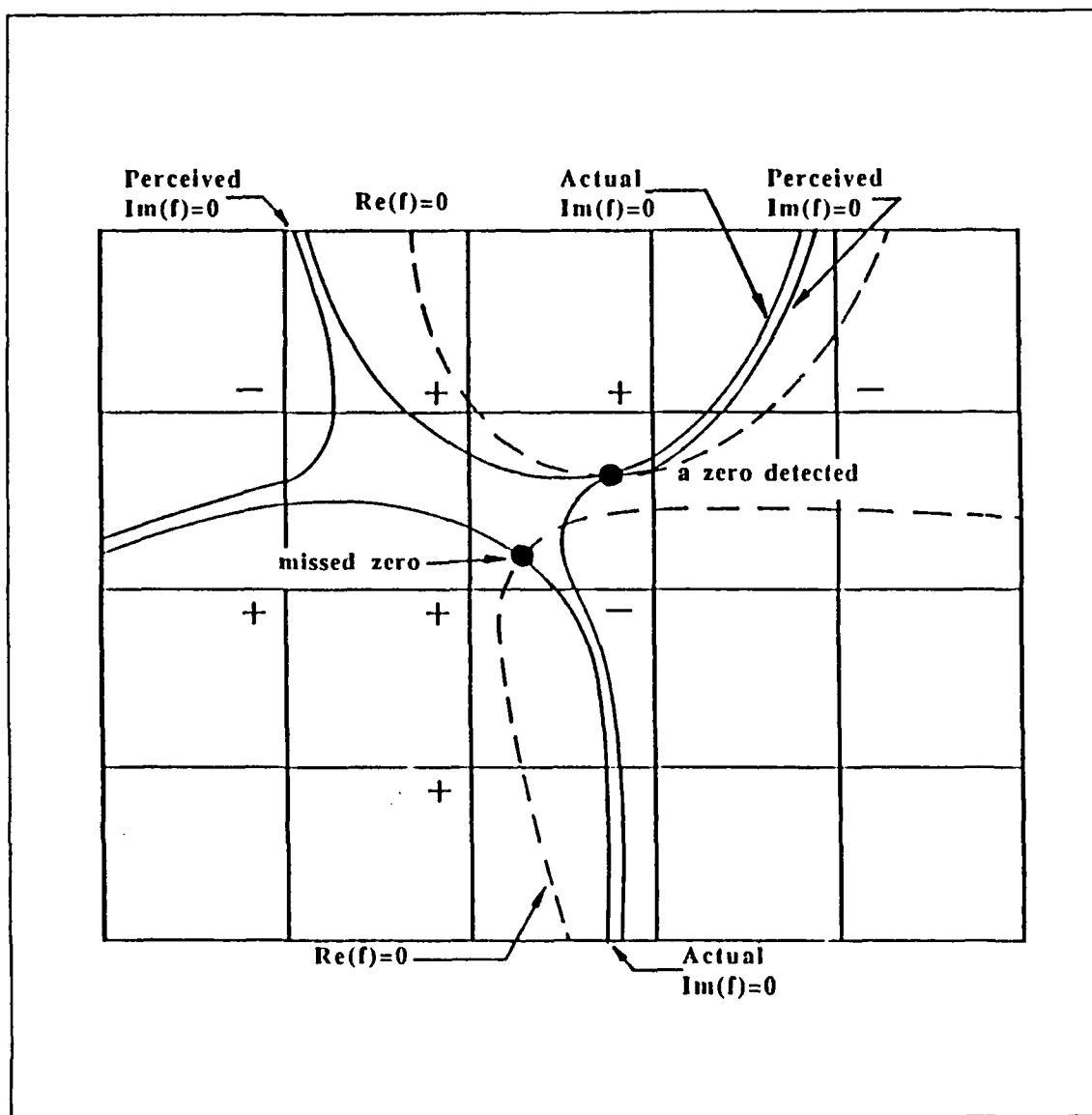


Figure 8. Crossing of two  $\text{Im}(f)=0$  at the same edge of the mesh square

The program (see subroutine *fzerox* in Chapter V) also checks for the presence of more than one zero on the same phase line. Such a situation may occur as in Figure 9. The program treats such an occurrence as an error and will proceed to reduce the mesh size. The root-search will start all over again.

One can infer that the presence of more than one zero on the same phase line is due to the inability to resolve two phase lines crossing a mesh square at the same edge. These

zeros actually lie on different phase lines and are 'legitimate' zeros. It is the author's opinion that these zeros are acceptable and a mesh-size reduction accompanied by a re-run of the root search routine unnecessary.

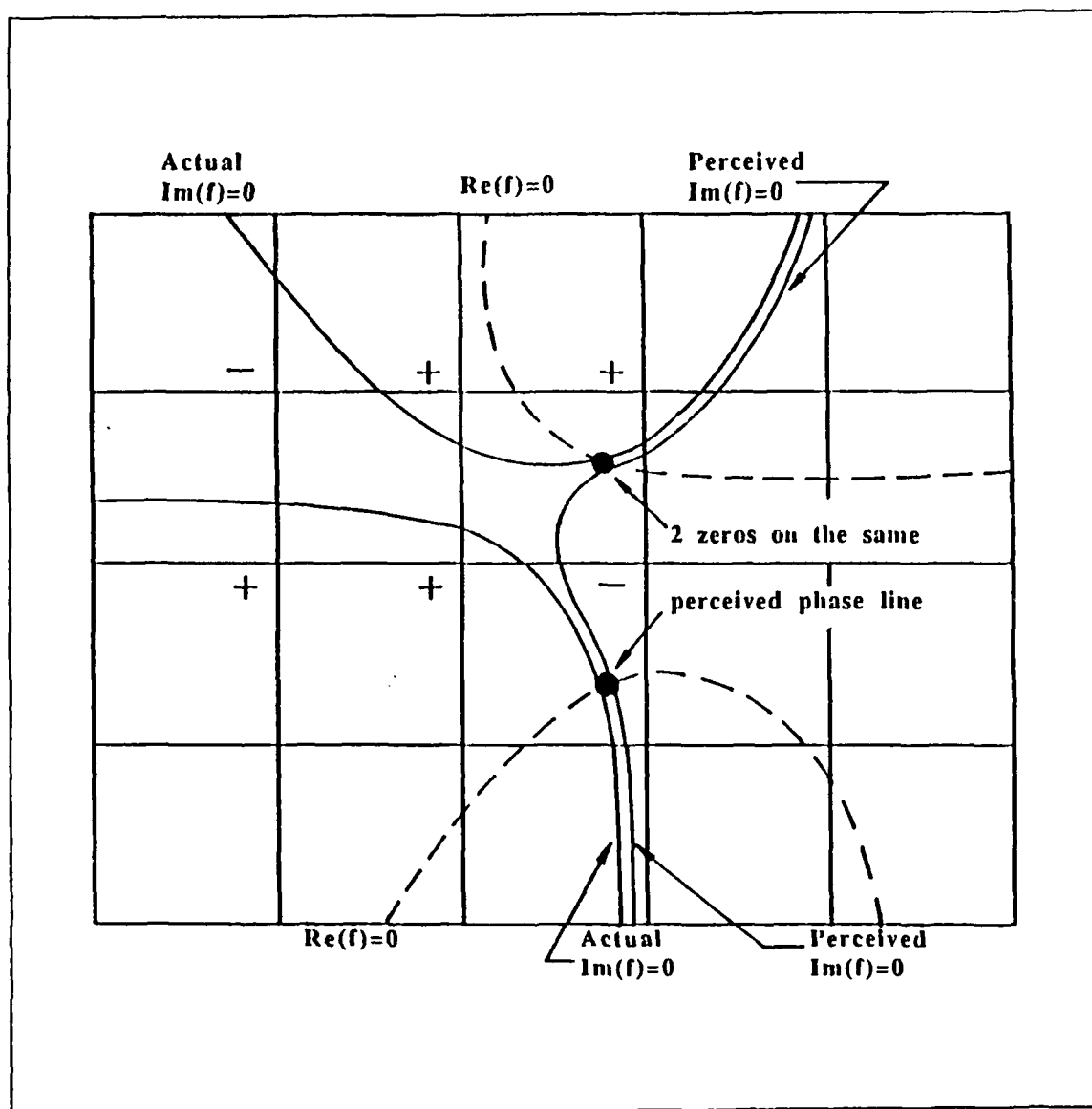


Figure 9. Two zeros found on the same phase line

The line  $\text{Im}(f)=0$  is traced from mesh to mesh until it exits the contour rectangle. For the example in Figure 6, the curve AB is followed through mesh squares 2, 3, 7, 11, 10, 14 and 13. In mesh square 11, both  $\text{Im}(f)=0$  and  $\text{Re}(f)=0$  pass through the mesh

square. The point of intersection can be determined from Equation 257. The mathematical details are presented in section C.

The contour AB is followed out of mesh square 11 into mesh squares 10 and 14. In mesh square 14, there are two  $\text{Im}(f)=0$  curves and one  $\text{Re}(f)=0$  curve. A crossing point is chosen to be a zero of  $f(z)$  if it lies on the  $\text{Im}(f)=0$  curve currently being followed. Therefore, when the contour AB is being traced, the zero in mesh square 14 will be ignored. However, this zero will be located when contour CD is being traced.

When contour AB exits the search rectangle at point B, the value of the leading edge (counterclockwise) of the mesh square is noted. In this example, the value of the leading edge is  $f^{(13)}(0, 1)$ . Identifying the exit points would prevent subsequent re-entering of the search rectangle via these points.

After the contour AB exits at B, the search for zeros of  $f(z)$  is continued by looking for more crossings of  $\text{Im}(f)=0$  with the search rectangle. Since mesh square 2 was the last mesh square that was checked for such crossings, the next mesh square to be examined will be mesh square 3.

The next crossing of  $\text{Im}(f)=0$  with the search rectangle is at mesh square 12. This contour CD is again followed through to mesh 21 where the exit point D is noted. The zero along CD in mesh square 14 is considered as an acceptable zero of  $f(z)$  since it lies on the contour CD which is the current contour being traced. The rectangular contour search will continue as described above until the search gets to mesh square 1, edge 4 where it stops.

### C. DETERMINATION OF THE COORDINATES OF ROOTS

The root of  $f(z)$  is the point of intersection of  $\text{Im}(f)=0$  and  $\text{Re}(f)=0$ . This can be determined from Equation 257.

By equating  $\text{Im}(f)$  of Equation 257 to zero, the following equations are obtained:

$$\gamma_k = - \frac{\text{Im}(a_k) + \text{Im}(b_k)\omega_k}{\text{Im}(c_k) + \text{Im}(d_k)\omega_k} \quad (264)$$

and

$$\omega_k = - \frac{\text{Im}(a_k) + \text{Im}(c_k)\gamma_k}{\text{Im}(b_k) + \text{Im}(d_k)\gamma_k} \quad (265)$$

Similarly, by equating  $\text{Re}(f)$  to zero,

$$\chi_k = - \frac{Re(a_k) + Re(b_k)\omega_k}{Re(c_k) + Re(d_k)\omega_k} \quad (266)$$

and

$$\omega_k = - \frac{Re(a_k) + Re(c_k)\chi_k}{Re(b_k) + Re(d_k)\chi_k} \quad (267)$$

Note that Equations 264 and 265 are equivalent; so are Equations 266 and 267. By equating the right hand side of Equation 264 and that of Equation 266, the following quadratic equation in  $\omega_k$  results:

$$P_1 \omega_k^2 + Q_1 \omega_k + W_1 = 0 \quad , \quad (268)$$

where

$$P_1 = Re(b_k) Im(d_k) - Im(b_k) Re(d_k) \quad , \quad (269)$$

$$Q_1 = Re(a_k) Im(d_k) + Re(b_k) Im(c_k) - Im(a_k) Re(d_k) - Im(b_k) Re(c_k) \quad (270)$$

and

$$W_1 = Re(a_k) Im(c_k) - Im(a_k) Re(c_k) \quad . \quad (271)$$

Similarly, from Equation 265 and 267, the quadratic equation in  $\chi_k$  is obtained:

$$P_2 \chi_k^2 + Q_2 \chi_k + W_2 = 0 \quad , \quad (272)$$

where

$$P_2 = Re(c_k) Im(d_k) - Im(c_k) Re(d_k) \quad , \quad (273)$$

$$Q_2 = Re(a_k) Im(d_k) + Re(c_k) Im(b_k) - Im(a_k) Re(d_k) - Im(c_k) Re(b_k) \quad (274)$$

and

$$W_2 = Re(a_k) Im(b_k) - Im(a_k) Re(b_k) \quad . \quad (275)$$

According to the program source code, if

$$|P_1| < |P_2| \quad , \quad (276)$$

then the root-finder routine uses Equations 264 and 268 to solve for  $z_s$  and  $\omega_s$ . If the condition in Equation 276 is not satisfied, then  $z_s$  and  $\omega_s$  will be determined from Equations 265 and 272.

However, it should be noted that if Equation 276 is true and if  $|P_1|$  approaches zero, one of the solutions of Equation 268 will become infinite. By the same token, if Equation 276 is not true and if  $|P_2|$  approaches zero, one of the solutions of Equation 272 will become infinite. The program (see subroutine quad in Chapter V) prevents such occurrences by incorporating a test such that the program returns only one solution when  $|P_1|$  or  $|P_2|$  approaches zero, i.e., Equation 268 or 272 reduces to a linear equation.

On the other hand, unless both  $|P_1|$  and  $|P_2|$  become very small, the above problem could be avoided if  $z_s$  and  $\omega_s$  are determined from Equations 265 and 272 when Equation 276 is true; otherwise Equations 264 and 268 are used. This modification is recommended.

A solution of Equations 265 and 272 or 264 and 268 will be valid if it lies within the current mesh square, i.e.,

$$0 \leq z_s \leq 1 \quad (277)$$

and

$$0 \leq \omega_s \leq 1 \quad , \quad (278)$$

where  $z_s = (z_s, \omega_s)$  is the solution.

In addition, the solution must also lie on the line  $\text{Im}(f)=0$  currently being traced. The test for this condition is derived below.

From Equation 257,  $\text{Im}(f)=0$  and  $\text{Re}(f)=0$  are equations of an equilateral hyperbola with vertical and horizontal asymptotes given by

$$z_c = -\frac{\text{Im}(b_s)}{\text{Im}(d_s)} \quad (279)$$



and

$$\omega_C = - \frac{\text{Im}(c)}{\text{Im}(d)} \quad (280)$$

Let  $(\chi_E, \omega_E)$  be the point at which  $\text{Im}(f)=0$  enter the mesh square. If

$$(\chi_E - \chi_C) (\chi_S - \chi_C) > 0 \quad (281)$$

and

$$(\omega_E - \omega_C) (\omega_S - \omega_C) > 0 \quad (282)$$

then the solution of  $f(z)$  lies on the current line.

Figure 10 illustrates the case where  $z_i$  is a proper solution and Figure 11 shows the case where  $z_i$  is not a proper solution of  $f(z)=0$ .

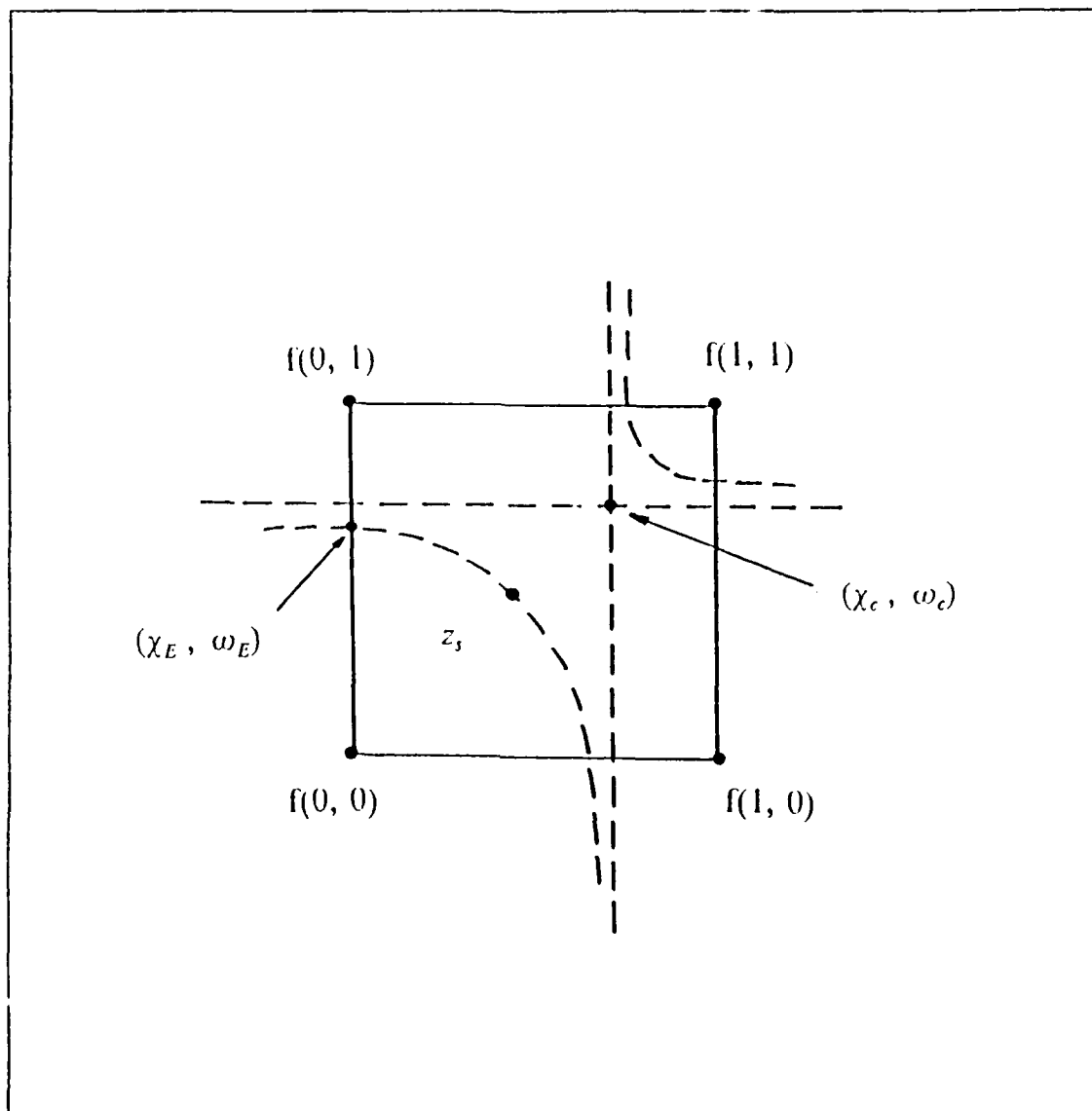


Figure 10.  $z_s$  is a proper solution of  $f(z) = 0$  [Alter Ref. 14]

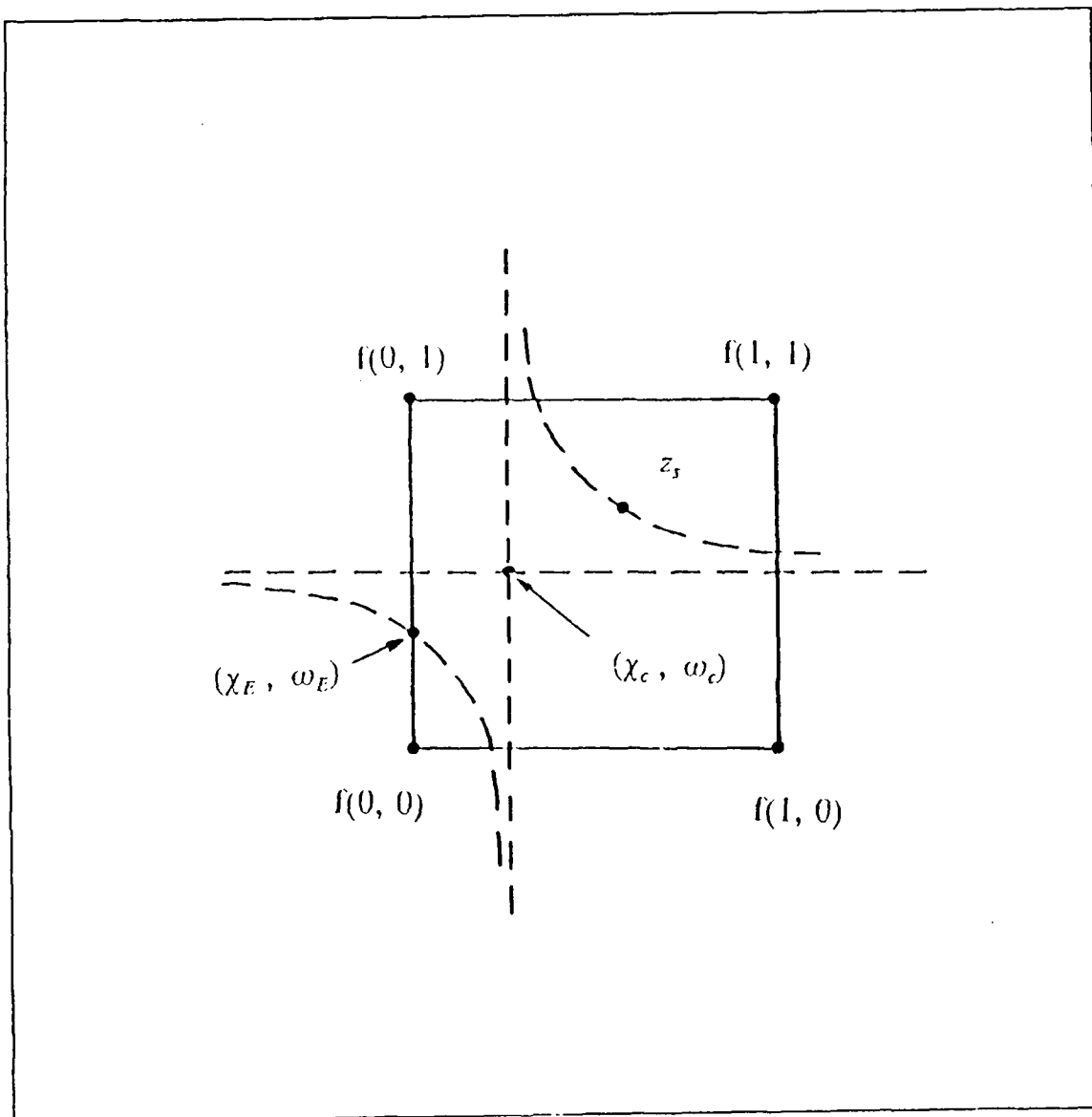


Figure 11.  $z_s$  is not a proper solution of  $f(z) = 0$  [After Ref. 14]

#### D. SOLUTION OF THE QUADRATIC EQUATION

Procedures for the solution of the quadratic equation are required for solving Equations 268 and 272.

Consider the quadratic equation of the form

$$ax^2 + 2bx + c = 0 \quad (283)$$

The solution is

$$\begin{aligned} x &= -\frac{2b \pm \sqrt{4b^2 - 4ac}}{2a} \\ &= \frac{b}{a} \left[ -1 \pm \sqrt{1 - \frac{ac}{b^2}} \right] \\ &= \frac{b}{a} \left[ -1 \pm \sqrt{1 + \varepsilon} \right] \end{aligned} \quad (284)$$

where

$$\varepsilon = -\frac{ac}{b^2} \quad (285)$$

Hence,

$$x_1 = \frac{b}{a} \left[ -1 + \sqrt{1 + \varepsilon} \right] \quad (286)$$

and

$$x_2 = \frac{b}{a} \left[ -1 - \sqrt{1 + \varepsilon} \right] \quad (287)$$

If  $\varepsilon \ll 1$ , then

$$(1 + \varepsilon)^{1/2} = 1 + \frac{1}{2}\varepsilon + \frac{(\frac{1}{2} - 1)}{2}\varepsilon^2 + \dots \quad (288)$$

Thus,

$$x_1 = \frac{h}{a} \left\{ \frac{1}{2} \left( -\frac{ac}{b^2} \right) + \frac{1}{2} \frac{\left( \frac{1}{2} - 1 \right)}{2} \left( -\frac{ac}{b^2} \right)^2 + \right. \\ \left. \frac{\frac{1}{2} \left( \frac{1}{2} - 1 \right) \left( \frac{1}{2} - 2 \right)}{3 \cdot 2} \left( -\frac{ac}{b^2} \right)^3 + \dots \right\} . \quad (289)$$

Factoring out  $\left( -\frac{ac}{b^2} \right)$ , Equation 289 becomes

$$x_1 = -\frac{1}{2} \frac{c}{b} \left\{ 1 + \frac{\left( \frac{1}{2} - 1 \right)}{2} \left[ -\frac{ac}{b^2} \right] \right. \\ \left. + \frac{\left( \frac{1}{2} - 1 \right) \left( \frac{1}{2} - 2 \right)}{3 \cdot 2} \left[ -\frac{ac}{b^2} \right]^2 + \dots \right\} . \quad (290)$$

Let

$$t_0 = 1 , \quad (291)$$

$$t_1 = \frac{\left( \frac{1}{2} - 1 \right)}{2} \left[ -\frac{ac}{b^2} \right] \quad (292)$$

and

$$t_2 = \frac{\left( \frac{1}{2} - 1 \right) \left( \frac{1}{2} - 2 \right)}{3 \cdot 2} \left[ \frac{ac}{b^2} \right]^2 . \quad (293)$$

In general,

$$t_k = t_{k-1} \left( \frac{k - \frac{1}{2}}{k + 1} \right) \left[ \frac{ac}{b^2} \right] \quad (294)$$

and

$$x_1 = -\frac{1}{2} \frac{c}{b} \{ t_0 + t_1 + t_2 + \dots + t_n \} \quad (295)$$

Once  $x_1$  is found,  $x_2$  can be found from Equations 286 and 287. By adding Equations 286 and 287,  $x_2$  is given by

$$x_2 = -2 \frac{b}{a} - x_1 \quad (296)$$

## E. NEWTON RAPHSON ITERATION

The zeros of  $f(z)$  found by the Shellman and Morfitt routine are only approximate solutions. The Newton Raphson iteration is used to improve the accuracy of the zero locations. The new root,  $z_{i+1}$ , is related to the previous root,  $z_i$ , by

$$z_{i+1} = z_i + \Delta z_i \quad (297)$$

where

$$\Delta z_i = -\frac{f(z_i)}{f'(z_i)} \quad (298)$$

Iteration of Equation 297 is continued until  $\Delta z_i$  is smaller than a pre-assigned tolerance.

## IV. SOLUTION OF STOKES' DIFFERENTIAL EQUATION

### A. STOKES' EQUATION

When the modified refractivity profile is modelled by a series of piecewise linear segments in the mathematical formulation of wave propagation in a multilayer tropospheric waveguide environment, the height gain function in each layer can be obtained from solution of Stokes' Equation.

$$\left[ \frac{\hat{r}^2}{\hat{c} q_i^2} + q_i \right] k_n(q_i) = 0 \quad ; \quad n = 1, 2 \dots \quad (299)$$

Solutions of Stokes' Equation are commonly given in terms of Airy functions or in terms of modified Hankel functions of order one-third. Several possible solutions exist and the choice of solution is dependent on numerical and physical considerations. Since  $Im(q_i)$  is proportional to the attenuation rate of the electromagnetic signal, physical considerations dictate that  $Im(q_i) \geq 0$ . In this program, the height gain function is expressed as a linear combination of  $k_1(q_i)$  and  $k_2(q_i)$  which are linearly independent for  $Im(q_i) \geq 0$ . The choice of  $k_1(q_i)$ ,  $k_2(q_i)$  or equivalently  $\{Ai(-q_i e^{-2\pi/3}), Ai(-q_i)\}$  is to assure that at least one of the two functions do not become very small numerically in the upper half of the complex  $q_i$ -plane. For the top layer, i.e.,  $i=1$ , the boundary condition at  $z \rightarrow \infty$  requires that the height gain function represents an outgoing wave. Hence in the topmost layer,  $h_2(q_i)$  or equivalently  $Ai(-q_i e^{-2\pi/3})$  is chosen to satisfy the radiation condition.

### B. EVALUATION OF AIRY FUNCTIONS

#### 1. General Outline

The algorithm for evaluation of complex Airy functions is adopted from Z. Schulten and D.G.M. Anderson [Ref. 16]. Evaluation of the complex function over the entire  $z$ -plane was achieved with a Taylor series and generalised Gaussian quadrature method. A Taylor series is used only for a small region near the origin within an ellipse with foci  $(-1.58, 3.95)$  and  $(0.28, -2.11)$ , major axis of 9.0 and minor axis of 6.4. Outside the ellipse, the Gaussian quadrature approximation is used. The elliptical boundary was chosen to satisfy restrictions imposed by computational methods.

Analysis of the Airy functions need only be done in the upper half complex plane as those in the lower half plane can be computed by the following conjugate relations:

$$Ai(z) = Ai^*(z^*) \quad (300)$$

and

$$\frac{dAi(z)}{dz} = \frac{dAi^*(z^*)}{dz} \quad (301)$$

Therefore, an algorithm can be developed to compute  $Ai(z)$  for  $0 \leq \arg z \leq 2\pi/3$  and by conjugacy,  $Ai(z)$  for  $-2\pi/3 \leq \arg z \leq 0$  can be computed.

For  $2\pi/3 < \arg z \leq 4\pi/3$ , the following connection formula of Airy functions can be used:

$$Ai(z) = e^{j\pi/3} Ai(ze^{-j2\pi/3}) + e^{-j\pi/3} Ai(ze^{j2\pi/3}) \quad (302)$$

## 2. Evaluation by Taylor Series Expansion

For small values of  $|z|$ , the Airy functions are evaluated by the series expansion [Ref. 1 and 4].

$$Ai(z) = C_1 f(z) + C_2 g(z) \quad (303)$$

and

$$Ai'(z) = C_1 f'(z) + C_2 g'(z) \quad (304)$$

where

$$f(z) = \sum_{k=0}^{\infty} z^k \frac{\Gamma\left(k + \frac{1}{3}\right)}{\Gamma\left(\frac{1}{3}\right)} \frac{z^{3k}}{(3k)!} \quad (305)$$



$$f'(z) = \sum_{k=1}^{\infty} 3^k \frac{\Gamma\left(k + \frac{1}{3}\right)}{\Gamma\left(\frac{1}{3}\right)} \frac{z^{3k-1}}{(3k-1)!} , \quad (306)$$

$$g(z) = \sum_{k=0}^{\infty} 3^k \frac{\Gamma\left(k + \frac{2}{3}\right)}{\Gamma\left(\frac{2}{3}\right)} \frac{z^{3k+1}}{(3k+1)!} , \quad (307)$$

$$g'(z) = \sum_{k=0}^{\infty} 3^k \frac{\Gamma\left(k + \frac{2}{3}\right)}{\Gamma\left(\frac{2}{3}\right)} \frac{z^{3k}}{(3k)!} . \quad (308)$$

$$C_1 = \left[ 3^{2/3} \Gamma\left(\frac{2}{3}\right) \right]^{-1} \quad (309)$$

and

$$C_2 = \left[ 3^{1/3} \Gamma\left(\frac{1}{3}\right) \right]^{-1} . \quad (310)$$

The number of terms required for convergence of the series is determined by a tolerance factor,  $\varepsilon$ , such that

$$|a_n| < \varepsilon \left| \sum_{i=0}^{n-1} a_i \right| , \quad (311)$$

where

$$Ai(z) = \sum_{i=0}^n a_i . \quad (312)$$

Similar test for convergence is implemented for  $Ai'(z)$ . In the program,  $\varepsilon$  is taken to be the machine epsilon, i.e.,

$$\varepsilon = 2^{-37} . \quad (313)$$

### 3. Evaluation by Generalised Gaussian Quadrature Method

Integral expressions for  $Ai(z)$  is given by [Ref. 16]

$$Ai(z) = \frac{1}{2\sqrt{\pi}} z^{-1/4} e^{-\zeta} \zeta \int_0^{\infty} \frac{\rho(x) dx}{\zeta + x} \quad (314)$$

$$\text{for } |z| > 0, \quad |\arg z| < \frac{2\pi}{3} .$$

where

$$\zeta = \frac{2}{3} z^{3/2} \quad (315)$$

and

$$\rho(x) = \pi^{-1/2} 2^{-11/6} 3^{-2/3} x^{-2/3} e^{-x} Ai\left(\frac{3x}{2}\right) . \quad (316)$$

The function,  $\rho(x)$ , is non-negative and exponentially decreasing. The integral in Equation 314 can be evaluated by the generalised Gaussian quadratic approximation, i.e.,

$$\int_0^{\infty} \frac{\rho(x) dx}{\zeta + x} \sim \sum_{i=1}^n \frac{w_i}{\zeta + x_i} , \quad (317)$$

where  $w_i$  is the quadrature weight and  $x_i$  is the abscissa. Hence

$$Ai(z) \sim \frac{1}{2} \pi^{-1/2} z^{-1/4} e^{-\zeta} \zeta \sum_{i=1}^n \frac{w_i}{\zeta + x_i} \quad (318)$$

and

$$Ai'(z) \sim \frac{1}{2} \pi^{-1/2} z^{1/4} e^{-z} \left[ \left( \frac{5}{6} - z \right) \sum_{i=1}^n \frac{w_i}{(z + x_i)} - z \sum_{i=1}^n \frac{w_i}{(z + x_i)^2} \right] . \quad (319)$$

The quadrature weights,  $w_i$ , and abscissae,  $x_i$ , had been calculated by Z. Schulten et al. [Ref. 16] and are given in Tables II to IV. These tables had been used for computation of single precision values up to seven significant digits.

Table 2. NUMBER OF TERMS REQUIRED IN QUADRATURE FORMULA  
[AFTER REF. 16]

$z$	$\arg z$	$Ai(z)$
$ z  \geq 11$	$< \frac{2\pi}{3}$	2 terms
$5 <  z  < 11$	$< \frac{2\pi}{3}$	4 terms
$2.5 <  z  < 5$	$< \frac{\pi}{3}$	4 terms

Table 3. 4-TERM GENERALISED GAUSSIAN INTEGRATION FOR AIRY  
FUNCTIONS [FROM REF. 16]

$i$	$x_i$	$w_i$
1.	3.9198329554455091	4.7763903057577263(−05)
2.	1.6915619004823504	4.9914306432910959(−03)
3.	5.0275532467263018(−01)	8.6169846993840312(−02)
4.	1.9247060562015692(−02)	9.0879095845981102(−01)

Table 4. 2-TERM GENERALISED GAUSSIAN INTEGRATION FOR AIRY FUNCTIONS [FROM REF. 16]

$x_i$	$w_i$
1.0592469382112378	3.1927194042263958(-02)
3.6800601866153044(-02)	9.6807280595773604(-01)

### C. EXTENDED COMPLEX ARITHMETIC

The extended complex arithmetic was introduced by Baumgartner in XWVG to handle complex numbers of large magnitude to avoid overflow problem.

It was discussed in Section D of Chapter II that for large values of  $|q|$ , the magnitude of  $k_1(q)$  and  $k_2(q)$  may become exponentially large or exponentially small.

Numerical evaluation of the modal function of Equation 166 can easily yield complex numbers with magnitudes as large as  $10^{+1000}$  or as small as  $10^{-1000}$ . Numbers of these magnitudes are outside the numerical limits of most computers.

To overcome this problem, each complex function,  $k_1(q)$  (or  $k_2(q)$ ), is represented by

$$k_1 = \hat{k}_1 e^{\phi} \quad , \quad (320)$$

where  $\hat{k}_1$  is the complex amplitude and  $\phi$  is the real exponent. In the program,  $\hat{k}_1$  is "normalized" such that

$$e^{-1} \leq \max [ |Re(\hat{k}_1)| , |Im(\hat{k}_1)| ] \leq e^{+1} \quad (321)$$

and  $\phi$  has integer values. By evaluating the complex amplitude and real exponent separately, overflow or underflow problems are avoided. Functions expressed in the form as in Equation 320 are known as extended complex numbers. Numerical manipulation such as addition and multiplication involving extended complex numbers are performed in the following manner. If  $z_1 = \hat{z}_1 e^{j\phi_1}$  and  $z_2 = \hat{z}_2 e^{j\phi_2}$  are two extended complex number, then their product is

$$\begin{aligned} z &= z_1 z_2 \\ &= \hat{z} e^{j\phi} , \end{aligned} \quad (322)$$

where

$$\hat{z} = \hat{z}_1 \hat{z}_2 \quad (323)$$

and

$$\phi = \phi_1 + \phi_2 . \quad (324)$$

The sum of two extended complex numbers is given by

$$\begin{aligned} z &= z_1 + z_2 \\ &= \hat{z} e^{j\phi} , \end{aligned} \quad (325)$$

where

$$\hat{z} = \hat{z}_1 + \hat{z}_2 e^{j(\phi_2 - \phi_1)} ; \phi_1 \geq \phi_2 \quad (326)$$

and

$$\phi = \phi_1 ; \phi_1 \geq \phi_2 \quad (327)$$

or

$$\hat{z} = \hat{z}_2 + \hat{z}_1 e^{j(\phi_1 - \phi_2)} ; \phi_1 < \phi_2 \quad (328)$$

and

$$\phi = \phi_2 ; \phi_1 < \phi_2 . \quad (329)$$

## V. DESCRIPTION OF MLAYER PROGRAM ELEMENTS

A short description of the functions of each program element in MLAYER is provided in this Chapter. The program had been tested and ran successfully on the Microsoft FORTRAN version 5.0 compilers and Microsoft C compilers version 5.0. The maximum number of layers and modes are 35 and 150 respectively. These limits are the upper bounds for operation on personal computers with 640 kilobytes of main memory.

### A. MAIN

#### i. Description

Main is the controlling program element for MLAYER. It calls appropriate subroutines to calculate modal eigenvalues, mode sum, path loss and radio horizon distance. Main also calls subroutine wvgstdin to read in data from input file, filein. Three output files are opened, namely, filein.out, filein.eig and filein.plt.

Filein.out provides a documented output of input parameters and a list of modal eigenvalues found in each search rectangle. Filein.out also contains a list of eigenvalues in increasing order of real parts with the corresponding values in the  $\theta$ -plane and respective attenuation of each mode in decibels per kilometer.

Filein.eig is an exact replica of the input file except that the parameter, myfile, is set to one and it contains a list of eigenvalues found. Once filein.eig is created, it can be used as the input file to re-run MLAYER for various transmitter/receiver range and height configurations.

Filein.plt contains eight columns of data, namely range, transmitter height, receiver height, coherent mode sum, incoherent mode sum, coherent path loss, incoherent path loss and radio horizon distance. The data are arranged in this manner to facilitate graph plotting.

Samples of filein, filein.out, filein.eig and filein.plt are given in Appendix B. In addition, main also computes the following:

- The derivative of the modified refractivity with respect to  $z$  for each layer.
- The derivative of the atmospheric absorption with respect to  $z$  for each layer.
- The absorptivity and its derivative with respect to  $z$  for each layer.
- The derivative of  $q$  with respect to  $z$  for each layer.
- The modal attenuation rate for each mode.

- The coefficients required for computation of surface roughness.
- The eigenvalues in the  $\theta$ -plane. The eigenvalues in the  $\theta$ -plane are related to those in the  $q_{1,1}$ -plane by

$$\theta = \sin^{-1} \left[ \sqrt{1 - \beta^2} \right] . \quad (330)$$

The conversion to  $\theta$ -plane is done solely for comparison of results with other models where root-search was performed in the  $\theta$ -plane.

## 2. Inputs

Input data are read in by subroutine wvgstdin in the following order:

- filein : Pathname of input file.
- mfile : If mfile = 0, read input data and calculate eigenvalues.  
If mfile = 1, read input data and eigenvalues.
- fqmzin : Initial frequency in megahertz.
- delfq : Frequency increment in megahertz.
- nfreq : Number of frequencies for which modes will be found.
- mpol : If mpol = 0, wave is horizontally polarized.  
If mpol = 1, wave is vertically polarized.
- aloss : Maximum rate of attenuation in decibels per kilometer of modes that will be found.
- scatmp : Seawater temperature in degrees Celsius.
- seaslt : Seawater salinity in grams of salt per kilogram of seawater.
- iflgab : If iflgab = 0, calculate the height invariant atmospheric absorption.  
If iflgab = 1, read in array containing values of atmospheric absorption at layer boundaries.
- airtrap : Air temperature in degrees Celsius at reference height, zref.  
Airtmp is used for calculating the height invariant atmospheric absorption when iflgab = 0.
- rh : Relative humidity in percent at reference height. Used for calculating the height invariant atmospheric absorption when iflgab = 0.
- wgpm3 : Water concentration in grams per cubic meter at reference height.  
Used for calculating the height invariant atmospheric absorption when iflgab = 0.
- rmsbht : Rms bump height in meters of the sea surface.
- ztinit : Initial height of transmitter in meters.
- delzt : Height increment of transmitter in meters.
- nzt : Number of transmitter heights at which field strength is to be



computed.

- **zrinit** : Initial height of receiver in meters.
- **delzr** : Height increment of receiver in meters.
- **nzr** : Number of receiver heights at which field strength is to be computed.
- **xinit** : Initial range separation of the transmitter and receiver in kilometers.
- **delx** : Increment in range separation in kilometers.
- **nx** : Number of range separation values at which field strength is to be computed.
- **zref** : Reference height at which air temperature, relative humidity and water concentration are specified.
- **nzlayr** : Number of linear segments used to model the modified refractivity profile.
- **zi(j)** : Array containing values of heights in meters at layer boundaries.
- **zim(j)** : Array containing values of modified refractivity at layer boundaries.
- **zigab(j)** : Array containing values of atmospheric absorption in decibels per kilometer at layer boundaries.

### 3. Outputs

The outputs to filein.out (logical file 16) are:

- **nrmode** : Number of modes found.
- **qeigen(j)** : Array containing values of modal eigenvalues found in the complex  $q_{1,1}$ -plane.
- **theta** : Complex eigenangle referenced to ground level for each mode.
- **atnu** : Rate of attenuation in decibels per kilometer for each mode.

The outputs to filein.eig (logical file 17) are :

- **nrmode** : Number of modes found.
- **qeigen(j)** : Array containing values of modal eigenvalues found in the complex  $q_{1,1}$ -plane.

The outputs to filein.plt (logical file 15)

- **ecms** : Coherent mode sum field strength relative to free space in decibels.
- **eims** : Incoherent mode sum field strength relative to free space in decibels.
- **ecpl** : Coherent mode sum path loss in decibels.
- **eipl** : Incoherent mode sum path loss in decibels.

#### 4. Calling subroutine

- None

#### 5. Subroutines called

- wvgstdin
- chkmod
- seah2o ; commented out
- findmod
- casin
- ao2h2o ; disabled
- modsum
- dhoriz
- function ibstrip

The call to subroutine seah2o was commented out in revision 3.0 of the program. To use the subroutine, remove the comment statement. Subroutine ao2h2o was disabled by setting iflgab to one. To enable ao2h2o, set iflgab to zero.

#### 6. Common block areas

- com 1
- com 2
- com 3
- pap 1
- datum

### B. SUBROUTINE ABCOEF

#### 1. Description

Subroutine abcoef calculates the coefficients,  $A_i$  and  $B_i$ , of the height gain function given by Equations 25 and 28. Computations of the coefficients are done in extended complex arithmetic. The normalizing integral of Equation 77 is also evaluated. The coefficients are then normalized by the following relation:

$$B_i = \frac{B_i}{\sqrt{\int_c f_{n_i}^2[q(z)]dz - j \frac{f_{n_i}'[q(0)]}{k^2} \left(\frac{k}{z_1}\right)^{2/3} \left(\frac{d_i}{dq_{1,1}}\right)}} \quad (331)$$

By noting that

$$\begin{aligned} \frac{d_i}{d\rho_m} &= \frac{d_i}{dq_{1,1}} \cdot \frac{dq_{1,1}}{d\rho_m} \\ &= -\frac{d_i}{dq_{1,1}} \cdot \left(\frac{k}{z_1}\right)^{2/3} \frac{2\rho_m}{k^2} \quad , \end{aligned} \quad (332)$$

it can be seen that the denominator of Equation 331 is identical to that of Equation 77.

In calculating the coefficients, the decision to integrate up or down is determined by the rate of attenuation given by

$$\text{att} = \text{Re}[\rho_m] \times 20 \log e \times 10^3 \quad \text{dB/km} \quad (333)$$

If the attenuation rate is greater than 0.1 dB km, upward integration is performed; otherwise downward integration is carried out.

At the same time, the subroutine also checks for evanescence using Equations 153 through 156 for upward integration and Equations 162 through 165 for downward integration.

## 2. Calling statement

The subroutine is called by the statement

Call abcoef (zero, m).

## 3. Inputs

The input variables are

- zero : An eigenvalue in the  $q_{1,1}$ -plane .
- m : Mode number.

## 4. Outputs

The output variables are

- acoefa(i,m) : Two dimensional array element of complex amplitude of A in the  $i^{\text{th}}$  layer and  $m^{\text{th}}$  mode.
- acoefe(i,m) : Two dimensional array element of real exponent of A in the  $i^{\text{th}}$  layer and  $m^{\text{th}}$  mode.

- $bcoefa(i,m)$ : Two dimensional array element of complex amplitude of B in the  $i^{th}$  layer and  $m^{th}$  mode.
- $bcoefe(i,m)$ : Two dimensional array element of real exponent of B in the  $i^{th}$  layer and  $m^{th}$  mode.

#### 5. Calling program element

- modsum

#### 6. Subroutines called

- xcdai
- xcadd

#### 7. Common block areas

- com 1
- com 2
- pap 1
- pap 2

### C. SUBROUTINE ADDX

#### 1. Description

This subroutine adds two extended real numbers according to the following equation:

$$z = z_1 + z_2 \quad . \quad (334)$$

where

$$z = z_a \exp(z_e) \quad . \quad (335)$$

$$z_1 = z_{1a} \exp(z_{1e}) \quad (336)$$

and

$$z_2 = z_{2a} \exp(z_{2e}) \quad . \quad (337)$$

#### 2. Calling statement

The subroutine is called by the statement

Call addx (za, ze, z1a, z1e, z2a, z2e).

### 3. Inputs

The input variables are

- z1a : Real amplitude of the extended real number,  $z_1$ .
- z1e : Real exponent of the extended real number,  $z_1$ .
- z2a : Real amplitude of the extended real number,  $z_2$ .
- z2e : Real exponent of the extended real number,  $z_2$ .

### 4. Outputs

The output variables are

- za : Real amplitude of the extended real number,  $z$ .
- ze : Real exponent of the extended real number,  $z$ .

### 5. Calling program element

- modsum

### 6. Subroutine called

- normre

### 7. Common block area

- none

## D. SUBROUTINE AO2H2O

### 1. Description

This subroutine calculates the atmospheric absorption coefficient in decibels per kilometer due to the quantum mechanical resonances of oxygen molecules and water vapour. Computed values are good for frequencies in the range 1 GHz to 1000 GHz.

### 2. Calling statement

The subroutine is called by the statement

Call ao2h2o (atmabs, fhz, tc, rh, zm, wgpm3).

### 3. Inputs

The input variables are

- fhz : Frequency in Hertz.
- tc : Temperature in degrees Celcius.
- rh : Relative humidity in percentage.
- zm : Height above sea level in meters.

- `wgpm3` : Water concentration in grams per cubic meter at reference height.

#### 4. Output

The output variable is

- `atmabs` : Atmospheric absorption coefficient in decibels per kilometer.

#### 5. Calling program element

- `main`

#### 6. Subroutine called

- `none`

#### 7. Common block area

- `none`

### E. SUBROUTINE CHKMOD

#### 1. Description

This subroutine checks for modal eigenvalues with zero value and discard them. A zero-value eigenvalue may sometimes lead to a divide-by-zero error in the mode sum calculation.

#### 2. Calling statement

The subroutine is called by the statement

Call `chkmod (qeigen, nrmode)`.

#### 3. Inputs

The input variables are

- `qeigen(j)` : Array containing location of modal eigenvalues in the complex  $q_{1,1}$ -plane.
- `nrmode` : Number of modes.

#### 4. Outputs

The output variables are

- `qeigen(j)` : Array containing location of non-zero eigenvalues in complex  $q_{1,1}$ -plane.
- `nrmode` : Number of modes.

**5. Calling program element**

- main

**6. Subroutine called**

- none

**7. Common block area**

- none

**F. SUBROUTINE DETR**

**1. Description**

This subroutine evaluates the modal function determinant of Equation 47 in extended complex arithmetic. The algorithm is discussed in Section E of Chapter II.

**2. Calling statement**

The subroutine is called by the statement

Call detr (deta, dete, nzlayr, d, de, dp1, dp1e, dm1, dm1e, dp2, dp2e, dm2, dm2e).

**3. Inputs**

The input variables are

- nzlayr : Number of linear segments used to model the modified refractivity profile.
- d : Array containing complex amplitudes of elements along the main diagonal of the modal function determinant.
- de : Array containing real exponents of elements along the main diagonal of the modal function determinant.
- dp1 : Array containing complex amplitudes of elements along the diagonal one above the main diagonal of the modal function determinant.
- dp1e : Array containing real exponents of elements along the diagonal one above the main diagonal of the modal function determinant.
- dm1 : Array containing complex amplitudes of elements along the diagonal one below the main diagonal of the modal function determinant.
- dm1e : Array containing real exponents of elements along the diagonal one below the main diagonal of the modal function determinant.
- dp2 : Array containing complex amplitudes of elements along the diagonal two above the main diagonal of the modal function determinant.
- dp2e : Array containing real exponents of elements along the diagonal two above the main diagonal of the modal function determinant.
- dm2 : Array containing complex amplitudes of elements along the diagonal two below the main diagonal of the modal function determinant.
- dm2e : Array containing real exponents of elements along the diagonal two

below the main diagonal of the modal function determinant.

#### 4. Outputs

The output variables are

- deta : Complex amplitude of the modal function.
- dete : Real exponent of the modal function.

#### 5. Calling program element

- fctvtx

#### 6. Subroutine called

- xcadd

#### 7. Common block area

- none

### G. SUBROUTINE DHORIZ

#### 1. Description

The subroutine calculates the radio horizon distance between the transmitter and receiver. The radio horizon distance in kilometers is given by

$$d = \sqrt{2} \cdot \sqrt{\frac{4a}{3}} [\sqrt{z_R} + \sqrt{z_T}] 10^{-3} \quad (338)$$

where  $a$  is the radius of the earth, and  $z_R$  and  $z_T$  are the receiver and transmitter heights respectively.

#### 2. Calling statement

The subroutine is called by the statement

Call dhoriz (dhz, zrcvr, zxmtr).

#### 3. Inputs

The input variables are

- zrcvr : Receiver height in meters.
- zxmtr : Transmitter height in meters.

#### 4. Output

The output variable is

- dhz : Radio horizon distance in kilometers.



### 5. Calling program element

- main

### 6. Subroutine called

- none

### 7. Common block area

- none

## H. SUBROUTINE DXDETR

### 1. Description

This subroutine evaluates the modal function determinant and its derivative in extended complex arithmetic. Values of the modal function determinant and its derivative are required by the Newton-Raphson root-finding routine. The methods for evaluating the modal function determinant and its derivative are discussed in Section E of Chapter II.

### 2. Calling statement

The subroutine is called by the statement

Call dxdetr (deta, dete, dxdeta, dxdete, nzlayr, d, de, dp1, dp1e, dm1, dm1e, dp2, dp2e, dm2, dm2e, dxd, dxde, dxdp1, dxdp1e, dxdm1, dxdm1e, dxdp2, dxdp2e, dxdm2, dxdm2e).

### 3. Inputs

The input variables are

- nzlayr : Number of linear segments used to model the modified refractivity profile.
- d : Array containing complex amplitudes of elements along the main diagonal of the modal function determinant.
- de : Array containing real exponents of elements along the main diagonal of the modal function determinant.
- dp1 : Array containing complex amplitudes of elements along the diagonal one above the main diagonal of the modal function determinant.
- dp1e : Array containing real exponents of elements along the diagonal one above the main diagonal of the modal function determinant.
- dm1 : Array containing complex amplitudes of elements along the diagonal one below the main diagonal of the modal function determinant.
- dm1e : Array containing real exponents of elements along the diagonal one below the main diagonal of the modal function determinant.
- dp2 : Array containing complex amplitudes of elements along the diagonal two above the main diagonal of the modal function determinant.

- **dp2e** : Array containing real exponents of elements along the diagonal two above the main diagonal of the modal function determinant.
- **dm2** : Array containing complex amplitudes of elements along the diagonal two below the main diagonal of the modal function determinant.
- **dm2e** : Array containing real exponents of elements along the diagonal two below the main diagonal of the modal function determinant.
- **dxd** : Array containing complex amplitudes of the derivatives of the elements along the main diagonal of the modal function determinant.
- **dxde** : Array containing real exponents of the derivatives of the elements along the main diagonal of the modal function determinant.
- **dxdp1** : Array containing complex amplitudes of the derivatives of the elements along the diagonal one above the main diagonal of the modal function determinant.
- **dxdp1e** : Array containing real exponents of the derivatives of the elements along the diagonal one above the main diagonal of the modal function determinant.
- **dxdm1** : Array containing complex amplitudes of the derivatives of the elements along the diagonal one below the main diagonal of the modal function determinant.
- **dxdm1e** : Array containing real exponents of the derivatives of the elements along the diagonal one below the main diagonal of the modal function determinant.
- **dxdp2** : Array containing complex amplitudes of the derivatives of the elements along the diagonal two above the main diagonal of the modal function determinant.
- **dxdp2e** : Array containing real exponents of the derivatives of the elements along the diagonal two above the main diagonal of the modal function determinant.
- **dxdm2** : Array containing complex amplitudes of the derivatives of the elements along the diagonal two below the main diagonal of the modal function determinant.
- **dxdm2e** : Array containing real exponents of the derivatives of the elements along the diagonal two below the main diagonal of the modal function determinant.

#### 4. Outputs

The output variables are

- **deta** : Complex amplitude of the modal function.
- **dete** : Real exponent of the modal function.
- **dxdeta** : Complex amplitude of the derivative of the modal function.
- **dxdete** : Real exponent of the derivative of the modal function.

## 5. Calling program element

- fdfdtx

## 6. Subroutine called

- xcadd

## 7. Common block area

- none

# I. SUBROUTINE FCTVLX

## 1. Description

This subroutine calculates the values of the elements in the matrix of the modal function of Equation 47 and invokes subroutine detx to evaluate the modal function. The subroutine shifts the imaginary part of  $q_{1,1}$  by  $qshift$  so that the real axis will not fall on the mesh line of the search rectangle set up by the Shellman-Morfitt routine. This allows the zeros on the real axis to be located. It is known that zeros near or on the negative real axis contribute significantly to the fields within a duct.

However, the shift implies that the Shellman-Morfitt routine will search for zeros of  $\|\Delta(t_{1,1})\|$  instead of  $\|\Delta(q_{1,1})\|$  where

$$t_{1,1} = q_{1,1} - j qshift \quad (339)$$

To compensate for the error, the shift back to  $q_{1,1}$  is done in subroutine fndmod for all zeros found.

## 2. Calling statement

The subroutine is called by the statement

Call fctvlx (q1lin, deta, dete).

## 3. Input

The input variable is

- q1lin : Location in the complex  $q_{1,1}$ -plane where the modal function is to be evaluated.

## 4. Outputs

The output variable are

- deta : Complex amplitude of the modal function.
- dete : Real exponent of the modal function.

## 5. Calling program element

- fzerox

## 6. Subroutines called

- xcdai
- xcadd
- detr
- surf

## 7. Common block areas

- com1
- com2
- com3
- com4

# J. SUBROUTINE FDFDTX

## 1. Description

This subroutine calculates the values of the elements in the matrices of  $\|\Delta(q_{1,i})\|$  and  $\frac{\partial}{\partial q_{1,i}} \|\Delta(q_{1,i})\|$  and then invokes subroutine dxcdetr to evaluate the modal function and its derivative. The subroutine also shifts the imaginary part of  $q_{1,i}$  by  $qshift$  as in subroutine fctvix. It is similarly compensated by a "shift-back" in subroutine fndmod.

## 2. Calling statement

The subroutine is called by the statement

Call fdfdtx (q1lin, deta, dete, dxdeta, dxdete).

## 3. Input

The input variable is

- q1lin : Location in the complex  $q_{1,i}$ -plane where the modal function and its derivative is to be evaluated.

## 4. Outputs

The output variables are

- deta : Complex amplitude of the modal function.
- dete : Real exponent of the modal function.
- dxdeta : Complex amplitude of the derivative of the modal function.
- dxdete : Real exponent of the derivative of the modal function.

### 5. Calling program element

- fzerox

### 6. Subroutines called

- xcdai
- xcadd
- dxdetr
- surf

### 7. Common block areas

- com1
- com2
- com3
- com4

## K. SUBROUTINE FINDFX

### 1. Description

The incorporation of the effects of surface roughness into the mathematical model results in discontinuity of the modal function along the imaginary  $q_{1,i}$ -axis. The Shellman-Morsitt routine will fail if the edge of the search rectangle lies on the imaginary axis.

Subroutine findfx will determine if any edge of the search rectangle is on the imaginary axis. If so, a small *offset* of  $1 \times 10^{-4}$  is introduced. If the left edge is on the imaginary axis, then the modal function is evaluated at  $Re(q_{1,i}) = + offset$ . If the right edge is on the imaginary axis, the modal function is evaluated at  $Re(q_{1,i}) = - offset$ .

### 2. Calling statement

The subroutine is called by the statement

Call findfx (jr, ji, f, fe, tleft, tright).

### 3. Inputs

The input variables are

- jr : Real part of the mesh square coordinates in mesh units.
- ji : Imaginary part of the mesh squares coordinates in mesh units.
- tleft : Real part of  $q_{1,i}$  at the left edge of the search rectangle.
- tright : Real part of  $q_{1,i}$  at the right edge of the search rectangle.

#### 4. Outputs

The output variables are

- $f$  : Complex amplitude of the modal function.
- $fe$  : Real exponent of the modal function.

#### 5. Calling program element

- $fzerox$

#### 6. Subroutine called

- $fctvlx$

#### 7. Common block area

- $fmccom$

### L. SUBROUTINE FNDMOD

#### 1. Description

This subroutine sets up the rectangular region in the complex  $q_{\perp}$ -plane for the Shellman-Morfit root finding routine to locate the zeros of the modal function.

The initial search rectangle is set up with

- the left edge of rectangle,  $tleft = 0.0$ .
- the right edge of rectangle,  $tright = tleft + tstep$ , where  $tstep$  is the length of the rectangle.
- the top of the rectangle,  $ttop$  given by [Ref. 5]

$$ttop = \frac{2 \times 10^{-3} A_{loss}}{k 20 \log e} \operatorname{Re} \left[ \left( \frac{k}{\alpha_1} \right)^{2/3} \right] , \quad (340)$$

where  $A_{loss}$  is the maximum attenuation rate in decibels per kilometer of those modes to be found.

- the bottom of the rectangle,  $tbot = -tol$ , where  $tol$  is the tolerance to which zeros are to be located.

In order that the zeros on the real axis can be located, a small offset of  $qshift$  is introduced in subroutines  $fdldtx$  and  $fctvlx$  so as to avoid having a mesh line on the real axis. This offset is corrected in the subroutine  $findmod$  by the statement

$$zeros(k) = zeros(k) - j qshift . \quad (341)$$

After the initial rectangle has been searched for zeros of the modal function, a new search rectangle is set up to the left of the initial rectangle. If no zero is found in three consecutive search rectangles, the search for roots to the left of the initial rectangle

is stopped. Findmod next forms search rectangles to the right of the initial rectangles. If no zero is found in three consecutive search rectangles, the search for roots to the right of the initial rectangle is stopped.

Since subroutine fzerox extends the search rectangle by one mesh unit on all sides, findmod also checks for and eliminates all zeros outside the current search rectangle.

After all the zeros have been found, findmod sorts them in order of increasing real parts and stores them in the array called zeros.

There are two versions of findmod, namely, findmod20.for and findmod80.for. Findmod80 sets up a smaller mesh and takes longer time to find the modes. Findmod80 should be used only if findmod20 fails to find all modes.

## 2. Calling subroutine

The subroutine is called by the statement

Call findmod (qeigen, nrmode, dmdz, detadx, zim, nzlayr, aloss, waveno).

## 3. Inputs

The input variables are

- dmdz : Array containing values of  $\frac{dM}{dz}$ .
- detadz : Array containing values of  $\frac{d\eta}{dz}$ .
- zim : Array containing values of  $M(z)$  at layer boundaries.
- nzlayr : Number of linear segments used to model the modified refractivity profile.
- aloss : Maximum attenuation rate (in decibels per kilometer) of modes to be found.
- waveno : Wave number.

## 4. Outputs

The output variables are

- qeigen : Array containing the zeros of the modal function.
- nrmode : Actual number of modes found.

### 5. Calling program element

- main

### 6. Subroutine called

- fzerox

### 7. Common block areas

- com4
- com5

## M. SUBROUTINE FZEROX

### 1. Description

Subroutine fzerox is the main subroutine used for finding the zeros of a complex function using the Shellman-Morfitt routine as described in Chapter III. Fzerox will convert the coordinates of the search rectangle edges into mesh units and proceed with the search for sign changes of  $\text{Im}(f)=0$  along each edge. The maximum number of crossings of  $\text{Im}(f)=0$  allowed with any edge of the search rectangle is 100.

Fzerox allows a maximum number, maxnsq, of mesh squares to lie on any one phase curve,  $\text{Im}(f)=0$ . If this number is exceeded, the program will reduce the mesh size by one-half and start over. If the problem persists after the mesh size has been reduced by maxnt times, an error message will be written to the output file and the program stops. In the program, maxnq is set at four times the number of mesh squares along the longest side of the search rectangle.

Fzerox also checks for modes found on the same phase line. If so, the program will reduce the mesh size and start over. If the problem remains after mesh size has been reduced by  $2^{\text{maxnt}}$  times, an error message will be written to output file and the program will stop.

### 2. Calling statement

The subroutine is called by the statement

Call Fzerox (tleft, tright, tbot, ttop, tmsho, tol, mprint, zeros).

### 3. Inputs

The input variables are

- tleft : Value of the real part of  $z$  at the left edge of search rectangle.
- tright : Value of the real part of  $z$  at the right edge of search rectangle.
- tbot : Value of the imaginary part of  $z$  at the bottom edge of the search rectangle.



- **ttop** : Value of the imaginary part of  $z$  at the top edge of the search rectangle.
- **tmsho** : Initial size of the mesh square.
- **tol** : Tolerance to which zeros are to be found. Zeros located closer than  $\text{tol}$  cannot be distinguished.
- **mprint** : A flag for debugging output. If  $\text{mprint} = 0$ , no debugging will be given.  $\text{Mprint} = 1$  will activate debugging printout.

#### 4. Outputs

The output variables are

- **zeros** : Array containing the complex zeros of  $f(z)$  in the specified search rectangle in the complex  $z$ -plane.
- **nrz** : The actual number of complex zeros of  $f(z)$  found.

#### 5. Calling program element

- **findmod**

#### 6. Subroutines called

- **findfx**
- **quad**
- **nomshx**

#### 7. Common block areas

- **newmsh**
- **tmccom**

### N. SUBROUTINE HTGAIN

#### 1. Description

This subroutine evaluates the normalized height gain function given by Equation 77.

#### 2. Calling Statement

The subroutine is called by the statement

Call **htgain** (**htga**, **htge**, **zero**, **z**, **m**).

#### 3. Inputs

The input variables are

- **zero** : Modal eigenvalue in  $q_{1,1}$ -space for mode  $m$ .
- **z** : Height (in meters) above ground at which height gain is to be evaluated.

- m : Mode index.

#### 4. Outputs

The output variables are

- htga : Complex height gain factor in extended arithmetic.
- htge : Real height gain exponent in extended arithmetic.

#### 5. Calling program element

- modsum

#### 6. Subroutines called

- xcdai
- xcadd

#### 7. Common block areas

- com2
- pap1
- pap2

### O. SUBROUTINE MODSUM

#### 1. Description

This subroutine calculates the field strength relative to free space given by Equations 100 and 101.

#### 2. Calling statement

The subroutine is called by the statement

Call modsum (ecms, eims, xm, rngfac, zr, zt, qeigen, ncount).

#### 3. Inputs

The input variables are

- xm : Range in meters.
- rngfac : Range factor calculated in the main program.
- zr : Receiver height in meters.
- zt : Transmitter height in meters.
- qeigen : Array containing eigenvalues in  $q_{1,1}$ -space .
- ncount : Counter set in the main program to avoid unnecessary calculations of acoefa, acoefe, bcoefa and bcoefe. If ncount = 1, subroutine modsum will call subroutine abcoef, otherwise subroutine modsum will compute modesum only.

#### 4. Outputs

The output variables are

- `ecms` : Coherent field strength relative to free space.
- `eims` : Incoherent field strength relative to free space.

#### 5. Calling program element

- `main`

#### 6. Subroutines called

- `abcoef`
- `htgain`
- `xcadd`
- `addx`

#### 7. Common block areas

- `com1`
- `com2`
- `pap2`

### P. SUBROUTINE NOMSHX

#### 1. Description

Subroutine `nomshx` takes the approximate locations of the complex zeros of  $f(z)$  provided by `fzerox` and improves the accuracies of the zero locations using Newton-Raphson iteration.

#### 2. Calling statement

The subroutine is called by the statement .

Call `nomshx (tol, zeros, nrz)`.

#### 3. Inputs

The input variables are

- `tol` : The tolerance to which zeros are to be located. Zeros located closer than `tol` cannot be distinguished.
- `zeros` : Array containing the approximate locations of the complex zeros of  $f(z)$ .
- `nrz` : The number of complex zeros stored in array `zeros`.

#### 4. Output

The output variable is

- zeros : An array containing the iterated values of the complex zeros of  $f(z)$ .

#### 5. Calling program element

- fzerox

#### 6. Subroutine called

- fdfdtx

#### 7. Common block area

- Newmsh

### Q. SUBROUTINE NORME

#### 1. Description

This subroutine "normalizes" the complex extended numbers,  $z = z_a \exp(z_e)$  such that

$$\exp(-1.0) \leq \max[|Re(z_a)|, |Im(z_a)|] \leq \exp(1.0) \quad (3-42)$$

and  $z_e$  has integer values.

#### 2. Calling statement

The program is called by the statement

Call norme (za, ze).

#### 3. Inputs

The input variables are

- za : Complex amplitude of the extended complex number, z.
- ze : Real exponent of the extended complex number, z

#### 4. Outputs

The output variables are

- za : Complex amplitude of the "normalized" extended complex number, z.
- ze : Real exponent of the "normalized" extended complex number, z.

#### 5. Calling program elements

- xcadd
- xcdaif
- xcdaig

**6. Subroutine called**

- none

**7. Common block area**

- none

**R. SUBROUTINE NORMRE**

**1. Description**

This subroutine "normalizes" the real extended number,  $z = z_a \exp(ze)$ , such that

$$\exp(-1.0) \leq |z_a| \leq \exp(+1.0) \quad (343)$$

and  $ze$  has integer values.

**2. Calling statement**

The subroutine is called by the statement

Call normre (za, ze).

**3. Inputs**

The input variables are

- $z_a$  : Real amplitude of the real extended number,  $z$ .
- $ze$  : Real exponent of the real extended number,  $z$ .

**4. Outputs**

The output variables are

- $z_a$  : Real amplitude of the "normalized" real extended number,  $z$ .
- $ze$  : Real exponent of the "normalized" real extended number,  $z$ .

**5. Calling program element**

- addx

**6. Subroutine called**

- none

**7. Common block area**

- none

## S. SUBROUTINE QUAD

### 1. Description

This subroutine finds the roots of the quadratic equation of the form given by Equation 283. If  $|\epsilon|$  given in Equation 285 is less than 0.3, the roots are computed with Equations 295 and 296. Otherwise Equations 286 and 287 are used. In addition, if  $|\frac{a}{2b}| < 10^{-30}$ , subroutine quad returns only 1 solution given by Equation 295.

### 2. Calling statement

The subroutine is called by the statement

Call quad (a, b, c, sol, nrsol, mprint).

### 3. Inputs

The input variables are

- a : The coefficient of  $x^2$  in the quadratic equation.
- b : The coefficient of  $2x$  in the quadratic equation.
- c : The constant term in the quadratic equation.

### 4. Outputs

The output variables are

- sol : An array containing the real roots of the quadratic equation.
- nrsol : The number of real roots found.
- mprint : A flag for debugging. If mprint is not equal to zero, values of the coefficients a, b, and c will be written to file.out.

### 5. Calling program element

- fzerox

### 6. Subroutine called

- none

### 7. Common block area

- none

## T. SUBROUTINE SEAH2O

### 1. Description

This subroutine evaluates the real effective relative dielectric constant and the real effective conductivity of seawater as a function of temperature, salinity and frequency. The program is based on the equations presented in section G of Chapter II.

## 2. Calling statement

The subroutine is called by the statement

Call seah2o (sigeff, epsell, t, s, freq).

## 3. Inputs

The input variables are

- t : Temperature of seawater in degree Celcius.
- s : Salinity of seawater in grams of salt per kilogram of seawater.
- freq : Frequency in Hertz.

## 4. Outputs

The output variables are

- sigeff : Real effective conductivity in siemens per meter.
- epsell : Real effective relative dielectric constant.

## 5. Calling program element

- main

## 6. Subroutine called

- none

## 7. Common block area

- none

# U. SUBROUTINE SURF

## 1. Description

This subroutine evaluates  $\gamma_z$  and  $\frac{d\gamma_z}{dq_{1,1}}$  when surface roughness is included in the model.

## 2. Calling statement

The subroutine is called by the statement

Call surf (q11, gamma, dgamdq).

## 3. Input

The input variable is

- q11 : Location in the complex  $q_{1,1}$ -plane where the modal function and its derivative are to be evaluated.

## 4. Outputs

The output variables are

- gamma : This is  $\gamma_z$  given by Equation 180 and 192 for horizontal and vertical

polarization respectively.

- `dgamdq` : This is  $\frac{d\epsilon}{dq_{\text{in}}}$  given by Equations 187 and 193 for horizontal and vertical polarization respectively.

#### 5. Calling program elements

- `fcvtx`
- `fdfdtx`

#### 6. Subroutines called

- function `ctanh`
- function `csech2`

#### 7. Common block areas

- `com1`
- `com5`
- `com6`

### V. SUBROUTINE XAINEG

#### 1. Description

This subroutine evaluates the Airy function,  $Ai(z)$ , and its derivative,  $Ai'(z)$ , for complex  $z$  in extended complex arithmetic for

$$-\frac{2\pi}{3} \leq \arg(z) \leq 0. \quad (344)$$

The subroutine will test if  $z^*$  is within the ellipse with foci  $(-158, 3.95)$  and  $(0.28, -2.11)$ , major axis of 9.0 and minor axis of 6.4. If  $z^*$  is within the ellipse, `xaineg` will invoke `xcdait` to compute the Airy function and its derivative with a Taylor series. Otherwise `xaineg` invokes `xcdaiG` which uses the Gaussian quadrature method.

#### 2. Calling statement

The subroutine is called by the statement

Call `xaineg` (`z`, `aia`, `aie`, `daia`, `daie`).

#### 3. Input

The input variable is

- `z` : A value in the complex  $z$ -plane at which the Airy function and its derivative are to be computed.



#### 4. Outputs

- aia : Complex amplitude of  $Ai(z)$  in extended complex arithmetic.
- aie : Real exponent of  $Ai(z)$  in extended complex arithmetic.
- daia : Complex amplitude of  $Ai'(z)$  in extended complex arithmetic.
- daie : Real exponent of  $Ai'(z)$  in extended complex arithmetic.

#### 5. Calling program element

- xcdai

#### 6. Subroutines called

- xcdait
- xcdaiq

#### 7. Common block area

- none

### W. SUBROUTINE XAIPOS

#### 1. Description

This subroutine evaluates the Airy function,  $Ai(z)$ , and its derivative,  $Ai'(z)$  in extended complex arithmetic for

$$0 \leq \arg(z) \leq \frac{2\pi}{3} . \quad (345)$$

The subroutine will test if  $z$  is within the ellipse with foci  $(-1.58, 3.95)$  and  $(0.28, -2.11)$ , major axis of 9.0 and minor axis of 6.4. If  $z$  is within ellipse, xaipos will invoke xcdait to compute the Airy function and its derivative with a Taylor series. Otherwise xaipos invokes xcdaiq which uses the Gaussian quadrature method.

#### 2. Calling statement

The subroutine is called by the statement

Call xaipos (z, aia, aie, daia, daie).

#### 3. Input

The input variable is

- z : A value in the complex  $z$ -plane at which the Airy function and its derivative are to be computed.

#### 4. Outputs

The output variables are

- aia : Complex amplitude of  $Ai(z)$  in extended complex arithmetic.
- aie : Real exponent of  $Ai(z)$  in extended complex arithmetic.
- daia : Complex amplitude of  $Ai'(z)$  in extended complex arithmetic.
- daie : Real exponent of  $Ai'(z)$  in extended complex arithmetic.

#### 5. Calling program element

- xcdai

#### 6. Subroutines called

- xcdait
- xcdaijg

#### 7. Common block area

- none

### X. SUBROUTINE XCDAI

#### 1. Description

The subroutine is the driving program in the evaluation of the Airy function,  $Ai(z)$ , and its derivative,  $Ai'(z)$ . If

$$0 < \arg(z) \leq \frac{\pi}{2}, \quad (346)$$

xcdai invokes xaipos to compute  $Ai(z)$  and  $Ai'(z)$  and invokes xaineg to compute  $Ai(ze^{-i\pi/3})$  and  $Ai'(ze^{-i\pi/3})$ . It then uses the connection formula of Airy function in Equation 302 and its derivative to compute  $Ai(ze^{i\pi/3})$  and  $Ai'(ze^{i\pi/3})$ .

If

$$-\frac{\pi}{2} \leq \arg(z) \leq 0, \quad (347)$$

xcdai invokes xaipos to compute  $Ai(ze^{i\pi/3})$  and  $Ai'(ze^{i\pi/3})$ . It also invokes xaineg to compute  $Ai(z)$  and  $Ai'(z)$ . It then uses the connection formula to compute  $Ai(ze^{-i\pi/3})$  and  $Ai'(ze^{-i\pi/3})$ .

If

$$\frac{2\pi}{3} < \arg(z) \leq \pi \quad (348)$$

or

$$-\pi \leq \arg(z) < -\frac{2\pi}{3} \quad , \quad (349)$$

xedai calls xaipos to compute  $Ai(ze^{-i2\pi/3})$  and  $Ai'(ze^{-i2\pi/3})$ . It also calls xaineg to compute  $Ai(ze^{i2\pi/3})$  and  $Ai'(ze^{i2\pi/3})$ . It then uses the connection formula to compute  $Ai(z)$  and  $Ai'(z)$ .

If

$$\frac{\pi}{2} < \arg(z) \leq \frac{2\pi}{3} \quad , \quad (350)$$

xedai calls xaipos to compute  $Ai(z)$  and  $Ai'(z)$  and calls xaineg to compute  $Ai(ze^{-i2\pi/3})$  and  $Ai'(ze^{-i2\pi/3})$ . It then uses the connection formula to compute  $Ai(ze^{i2\pi/3})$  and  $Ai'(ze^{i2\pi/3})$ .

If

$$-\frac{2\pi}{3} < \arg(z) < -\frac{\pi}{2} \quad , \quad (351)$$

xedai calls xaipos to compute  $Ai(ze^{i2\pi/3})$  and  $Ai'(ze^{i2\pi/3})$ . It also calls xaineg to compute  $Ai(z)$  and  $Ai'(z)$ . It then uses the connection formula to compute  $Ai(ze^{-i2\pi/3})$  and  $Ai'(ze^{-i2\pi/3})$ .

## 2. Calling statement

The subroutine is called by the statement

Call xedai (z, aia, aie, daia, daie, aipa, aipe, daipa, daipe, aima,  
aime, daima, daime).

## 3. Input

The input variable is

- z : A value in the complex z-plane at which the Airy function and its derivative are to be computed.

## 4. Outputs

The output variables are

- aia : Complex amplitude of  $Ai(z)$  in extended arithmetic.

- aie : Real exponent of  $Ai(z)$  in extended complex arithmetic.
- daia : Complex amplitude of  $Ai'(z)$  in extended complex arithmetic.
- daie : Real exponent of  $Ai'(z)$  in extended complex arithmetic.
- aipa : Complex amplitude of  $Ai(ze^{j2\pi/3})$  in extended complex arithmetic.
- aipe : Real exponent of  $Ai(ze^{j2\pi/3})$  in extended complex arithmetic.
- daipa : Complex amplitude of  $Ai'(ze^{j2\pi/3})$  in extended complex arithmetic.
- daipe : Real exponent of  $Ai'(ze^{j2\pi/3})$  in extended complex arithmetic.
- aima : Complex amplitude of  $Ai(ze^{-j2\pi/3})$  in extended complex arithmetic.
- aime : Real exponent of  $Ai(ze^{-j2\pi/3})$  in extended complex arithmetic.
- daima : Complex amplitude of  $Ai'(ze^{-j2\pi/3})$  in extended complex arithmetic.
- daime : Real exponent of  $Ai'(ze^{-j2\pi/3})$  in extended complex arithmetic.

#### 5. Calling program element

- fctvix

#### 6. Subroutines called

- xaipos
- xaineg
- xcadd

#### 7. Common block area

- none

### Y. SUBROUTINE XCDAIG

#### 1. Description

The subroutine computes the Airy function,  $Ai(z)$ , and its derivative,  $Ai'(z)$ , in extended arithmetic using the Gaussian quadrature method.

#### 2. Calling statement

The subroutine is called by the statement

Call xcdaig (z, aia, aie, daia, daie).

#### 3. Inputs

The input variables is

- z : A value in the complex z-plane at which the Airy function and its derivative are to be computed.

#### 4. Outputs

The output variables are

- aia : Complex amplitude of  $Ai(z)$  in extended arithmetic.
- aie : Real exponent of  $Ai(z)$  in extended complex arithmetic.
- daia : Complex amplitude of  $Ai'(z)$  in extended complex arithmetic.
- daie : Real exponent of  $Ai'(z)$  in extended complex arithmetic.

#### 5. Calling program elements

- xaipos
- xaineg

#### 6. Subroutine called

- norme

#### 7. Common block area

- norme

### Z. SUBROUTINE XCDAIT

#### 1. Description

The subroutine computes the Airy function,  $Ai(z)$ , and its derivative,  $Ai'(z)$ , in extended arithmetic using a Taylor series expansion.

#### 2. Calling statement

The subroutine is called by the statement

Call xcdaic (z, aia, aie, daia, daie).

#### 3. Input

The input variable is

- z : A value in the complex  $z$ -plane at which the Airy function and its derivative are to be computed.

#### 4. Outputs

The output variables are

- aia : Complex amplitude of  $Ai(z)$  in extended arithmetic.
- aie : Real exponent of  $Ai(z)$  in extended complex arithmetic.
- daia : Complex amplitude of  $Ai'(z)$  in extended complex arithmetic.
- daie : Real exponent of  $Ai'(z)$  in extended complex arithmetic.

### 5. Calling program elements

- xaipos
- xaineg

### 6. Subroutine called

- norme

### 7. Common block area

- none

## AA. FUNCTION PROGRAMS

### 1. Function casin

Function casin evaluates the complex arcsine of a complex number,  $z$ , with the equation

$$\sin^{-1}(z) = -j \ln [(1 - z^2)^{1/2} + jz] \quad (352)$$

The function is called by the statement

casin (z).

The function is used in the main program to convert the eigenvalues from  $q_{1,1}$ -plane to  $\theta$ -plane .

### 2. Function csech2

Function csech2 evaluates the square of the hyperbolic secant of a complex number,  $z$ , with the equation

$$\operatorname{sech}^2(z) = \frac{4}{e^{2z} + 2 + e^{-2z}} \quad (353)$$

The function is called by the statement

csech2 (z).

It is used in subroutine surf for evaluating  $\frac{\partial a}{\partial q_{1,1}}$ ,  $\frac{\partial b}{\partial q_{1,1}}$ ,  $\frac{\partial \tilde{a}}{\partial q_{1,1}}$  and  $\frac{\partial \tilde{b}}{\partial q_{1,1}}$  (see Section F, Chapter II).

### 3. Function ctanh

Function ctanh evaluates the hyperbolic tangent of a complex number,  $z$ , with the equation

$$\tanh(z) = \frac{e^z - e^{-z}}{e^z + e^{-z}}$$

$$\text{for } | \operatorname{Re}(z) | > \frac{1}{10\sqrt{2}} \quad (354)$$

$$\text{or } | \operatorname{Im}(z) | > \frac{1}{10\sqrt{2}}$$

or

$$\tanh(z) = z - \frac{1}{3} z^3 + \frac{2}{15} z^5 - \frac{17}{315} z^7 + \frac{6z}{2835} z^9 \dots$$

$$\text{for } | \operatorname{Re}(z) | \leq \frac{1}{10\sqrt{2}} \quad (355)$$

$$\text{or } | \operatorname{Im}(z) | \leq \frac{1}{10\sqrt{2}}$$

The function is called by the statement

`ctanh (z).`

It is used in subroutine `surf` for evaluating  $a$ ,  $b$ ,  $\tilde{a}$ ,  $\tilde{b}$  and their derivatives.

#### 4. Function `ibstrip`

Function `ibstrip` finds the length of a character string by removing all trailing spaces from the end of the string and returns the number of characters in the stripped string.

The function is called by the statement

`ibstrip (inst, outst).`

where `inst` is the input string and `outst` is the stripped string.

The function is used in the main program to check the length of the filename.

### AB. MLAYER SUPPORTING PROGRAMS

MLAYER has several supporting programs that helps in the running of the program. They are

- `mlayer.mak`
- `wvgclean.mak`
- `wvgstrip.mak`
- `wvgclean.c`

- wvgstrip.for
- wvgstdin.for

#### 1. Mlayer.mak

This is the makefile for the creation of mla20 and mla80 programs. Mla20 uses subroutine findmod20.for to set up the rectangular search region whereas mla80 uses findmod80.for.

Mlayer.mak compiles and links all program elements and creates mla20.exe and mla80.exe for the execution of MLAYER program with either mla20 or mla80.

To compile and link MLAYER, type

```
nmake mlayer.mak
```

Nmake will compare the modification dates of the target files (e.g., object and executable files) with those of the dependent files (e.g., source files). If any of the dependent files has been changed recently, i.e., the modification dates of the dependent files are more recent than the target files, nmake will execute the commands in mlayer.mak to update the target files. Nmake updates only those outdated target files.

To execute MLAYER program, type

```
mla20 < infile
```

to execute the mla20 version. Infile is the name of the input file which can be either filein or filein.eig. If filein is used, MLAYER will search for all modes of propagation. If filein.eig is used, MLAYER will compute the modsum and path loss without a search for the modes.

#### 2. Wvgclean.mak

This is the makefile for the creation of wvgclean.exe

#### 3. Wvgstrip.mak

This is the makefile for the creation of wvgstrip.exe.

#### 4. Wvgclean.c

Wvgclean.c adds comment or descriptors to each datum line in filein and filein.eig. This will help in editing.

To execute the program, type

```
wvgclean < filein.eig > temp  
ren temp filein.eig
```

#### 5. Wvgstrip.for

Wvgstrip.for removes comments or descriptors that are added by wvgclean.c.

To execute the program, type



```
wvgstrip < filein.eig > temp  
ren temp filein.eig
```

**6. Wvgstdin.for**

Wvgstdin.for reads input data from filein or filein.eig on execution of the MLAYER program.

## **VI. DISCUSSIONS AND RECOMMENDATIONS**

### **A. ASSESSMENT OF MLAYER**

MLAYER was developed with the intention of providing a program which is capable of locating all propagating modes with attenuation rates below a specified value. However, the price of this capability is the extremely long execution time. For instance, a 2-meter evaporation duct takes approximately 3 hours and 20 minutes (on an IBM PS 2 model 80 with an Intel 80386 processor at 16 MHz) to locate 9 modes with attenuation rates less than 5 decibels per kilometer. A 14-meter evaporation duct takes about 6 hours to locate 94 modes with attenuation rates below 2.1 decibels per kilometer. Samples of output data are found in Appendix B.

Calculations of field strength and path loss with MLAYER were found to agree favourably with experimental measurements [Ref. 17 and 18]. Typically, the model underestimates the measured values by about 10 dB. The sources of discrepancies were probably due to the validity of the surface roughness model and the assumption of lateral homogeneity of the medium.

In the following sections, proposed areas for enhancement of the capabilities of MLAYER are discussed.

### **B. SURFACE ROUGHNESS MODEL**

The surface roughness model used in MLAYER is based on the Kirchhoff approximation which is a single scattering theory. This model is only valid for gently undulating surfaces where shadowing of incident and scattered fields are negligible. When shadowing cannot be neglected, the Kirchhoff approximation tends to overpredict the scattered energy which leads to erroneous results.

Shadowing occurs when incoming or outgoing rays are blocked by the rough surface. This leaves some area of the surface in shadow. This effect is associated with multiple scattering. Therefore, to account for shadowing, multiple scattering interactions must be included in the model.

A more accurate model called the phase perturbation technique was introduced by Winebrenner and Ishimaru [Ref. 19]. The phase perturbation theory incorporates all the multiple scatterings required for shadowing of the incident field. The theory is based on the extinction theorem which is a consequence of Green's theorem [Ref. 20]. In Reference 19, numerical results of the phase perturbation bistatic scattering cross section were

compared with exact numerical results. In the region examined, it was found that the results of the phase perturbation technique agree with the exact results except at low grazing angles.

The phase perturbation technique is valid for a wider range of applications. It was shown in Reference 19 that the reflection and backscattering coefficients reduce to that of the Kirchhoff approximation when  $l/\lambda \gtrsim 1$ , i.e., for gently undulating surfaces, where  $l$  is the length and  $\lambda$  is the wavelength of the incident wave.

### **C. LATERAL HOMOGENEITY OF REFRACTIVITY PROFILE**

In MLAYER, the refractive index is assumed to vary only with height. Although this assumption appears to be adequate most of the time [Ref. 17], there are incidents where the validity of lateral homogeneity was cited as a possible reason for discrepancies between observations and predicted results [Ref. 18 and 21].

For refractivity profile that varies along the path, the laterally inhomogeneous structures can be approximated by several homogeneous sections. The exchange of energy between the propagating modes at the boundary of two sections are analyzed by mode-conversion techniques [Ref. 22].

### **D. FIELD CONTRIBUTION FROM THE SOURCE**

In the formulation of the propagation model, MLAYER neglects the contribution from the direct wave of the source. This approximation is good for ranges near or beyond the horizon where the major field contributions are from the reflected waves. For ranges well within the horizon, the contribution from the direct wave is significant and should not be ignored. Hence, for wave propagation in the interference region, the coefficients,  $A_1$  and  $B_1$ , of Equations 31 and 32 must be obtained from the solutions of Equation 46. At the same time, the surface wave field contribution from the integral of the contour,  $C_2$ , in Equation 73 may become significant.

## VII. CONCLUSION

MLAYER is a useful research tool for conducting case studies over a large dynamic range of frequencies. Calculation of field strength and path loss are in agreement with observations for frequencies from 63 Mhz to 94 GHz [Ref. 17 and 18]. It can serve as a yardstick against which results of other quicker but less accurate models such as IREPS [Ref. 23] and LREPS [Ref. 24] are compared. In addition, the multiple (as opposed to trilinear) piecewise linear refractivity profile makes it possible to model the simultaneous occurrence of an evaporation duct and elevated ducts.

Finally, modifications of MLAYER to incorporate conditions of horizontal heterogeneity of refractivity, effects of source contribution and an improved modelling of surface roughness as discussed in Chapter VI are recommended to fully exploit its capabilities. These are opportunities opened for further studies.

## APPENDIX A. REPRESENTATION OF $k_1(q)$ AND $k_2(q)$ IN TERMS OF MODIFIED HANKEL FUNCTION OF ORDER ONE-THIRD

The Airy function is related to the modified Bessel function of the third kind of order one-third by the following equation [Ref. 4]:

$$Ai(q) = \frac{1}{\pi} \left( \frac{q}{3} \right)^{1/2} K_{1/3} \left( \frac{2}{3} q^{3/2} \right), \quad (356)$$

where

$$K_{1/3}(q) = j \frac{\pi}{2} e^{j\pi/6} H_{1/3}^{(1)}(qe^{j\pi/2}) \quad (357)$$

or

$$K_{1/3}(q) = -j \frac{\pi}{2} e^{-j\pi/6} H_{1/3}^{(2)}(qe^{-j\pi/2}). \quad (358)$$

In Equations 357 and 358,  $H_{1/3}^{(1)}(.)$  and  $H_{1/3}^{(2)}(.)$  are Hankel functions of the first and second kind of order one-third respectively. Substituting Equation 357 into Equation 356 leads to

$$\begin{aligned} Ai(q) &= \frac{1}{\pi} \left( \frac{q}{3} \right)^{1/2} j \frac{\pi}{2} e^{j\pi/6} H_{1/3}^{(1)} \left( \frac{2}{3} q^{3/2} e^{j\pi/2} \right) \\ &= \frac{j}{2} \left( \frac{q}{3} \right)^{1/2} e^{j\pi/6} H_{1/3}^{(1)} \left( \frac{2}{3} q^{3/2} e^{j\pi/2} \right). \end{aligned} \quad (359)$$

Thus,

$$\begin{aligned} Ai(-qe^{j2\pi/3}) &= \frac{j}{2} \left( \frac{q}{3} \right)^{1/2} (e^{-j\pi})^{1/2} (e^{j2\pi/3})^{1/2} e^{j\pi/6} \\ &\quad \times H_{1/3}^{(1)} \left( \frac{2}{3} q^{3/2} e^{-j3\pi/2} e^{j\pi} e^{j\pi/2} \right) \\ &= \frac{j}{2} \left( \frac{q}{3} \right)^{1/2} H_{1/3}^{(1)} \left( \frac{2}{3} q^{3/2} \right). \end{aligned} \quad (360)$$

From Equation 142,

$$k_1(q) = -j 2(12)^{1/6} Ai(-q e^{j2\pi/3}) \quad (361)$$

Substituting Equation 360 into Equation 361 gives

$$\begin{aligned} k_1(q) &= (12)^{1/6} \left(\frac{q}{3}\right)^{1/2} H_{1/3}^{(1)}\left(\frac{2}{3} q^{3/2}\right) \\ &= \left(\frac{2}{3} q^{3/2}\right)^{1/3} H_{1/3}^{(1)}\left(\frac{2}{3} q^{3/2}\right) \\ &= h_1(q) \quad (362) \end{aligned}$$

where  $h_1(q)$  is the modified Hankel function of the first kind of order one-third and is defined as [Ref. 5]

$$h_1(q) \equiv \left(\frac{2}{3} q^{3/2}\right)^{1/3} H_{1/3}^{(1)}\left(\frac{2}{3} q^{3/2}\right) \quad (363)$$

To derive an expression for  $k_2(q)$ , the Airy function,  $Ai(-q)$ , can be expressed as [Ref. 4]

$$Ai(-q) = \frac{1}{2} \left(\frac{q}{3}\right)^{1/2} \left[ e^{j\pi/6} H_{1/3}^{(1)}\left(\frac{2}{3} q^{3/2}\right) + e^{-j\pi/6} H_{1/3}^{(2)}\left(\frac{2}{3} q^{3/2}\right) \right] \quad (364)$$

From Equation 143,

$$k_2(q) = 2(12)^{1/6} e^{j\pi/6} Ai(-q) \quad (365)$$

Substituting Equation 364 into 365 leads to

$$\begin{aligned}
k_z(q) &= (12)^{1/6} e^{j\pi/6} \left(\frac{q}{3}\right)^{1/2} \left[ e^{j\pi/6} H_{1/3}^{(1)}\left(\frac{2}{3} q^{3/2}\right) \right. \\
&\quad \left. + e^{-j\pi/6} H_{1/3}^{(2)}\left(\frac{2}{3} q^{3/2}\right) \right] \\
&= \left(\frac{2}{3} q^{3/2}\right)^{1/3} \left[ e^{j\pi/3} H_{1/3}^{(1)}\left(\frac{2}{3} q^{3/2}\right) + H_{1/3}^{(2)}\left(\frac{2}{3} q^{3/2}\right) \right] \quad (366) \\
&= \left(\frac{2}{3} q^{3/2}\right)^{1/3} H_{1/3}^{(2)}\left(\frac{2}{3} q^{3/2}\right) \\
&\quad - e^{j4\pi/3} \left(\frac{2}{3} q^{3/2}\right)^{1/3} H_{1/3}^{(1)}\left(\frac{2}{3} q^{3/2}\right) \\
&= h_2(q) - e^{j2\pi/3} h_1(q) \quad ,
\end{aligned}$$

where  $h_2(q)$  is the modified Hankel function of the second kind of order one-third and is defined as [Ref. 5]

$$h_2(q) \equiv \left(\frac{2}{3} q^{3/2}\right)^{1/2} H_{1/3}^{(2)}\left(\frac{2}{3} q^{3/2}\right) \quad . \quad (367)$$

For large  $|q|$ , the Hankel function can be approximated by the leading term of its asymptotic expansion. That is [Ref. 4],

$$\begin{aligned}
H_{1/3}^{(1)}(q) &\sim \left(\frac{2}{\pi q}\right)^{1/2} \exp\left[j\left(q - \frac{\pi}{6} - \frac{\pi}{4}\right)\right] \quad ; \\
&\quad -\pi < \arg(q) < 2\pi
\end{aligned} \quad (368)$$

and

$$\begin{aligned}
H_{1/3}^{(2)}(q) &\sim \left(\frac{2}{\pi q}\right)^{1/2} \exp\left[-j\left(q - \frac{\pi}{6} - \frac{\pi}{4}\right)\right] \quad ; \\
&\quad -2\pi < \arg(q) < \pi \quad .
\end{aligned} \quad (369)$$

Therefore,

$$\begin{aligned}
 k_1(q) &\sim 12^{1/6} \left(\frac{q}{3}\right)^{1/2} \left(\frac{2}{\pi}\right)^{1/2} \left(\frac{2}{3} q^{3/2}\right)^{-1/2} \exp \left[ j \left( \frac{2}{3} q^{3/2} - \frac{5\pi}{12} \right) \right] \\
 &\sim 12^{1/6} \pi^{-1/2} q^{-1/4} \exp \left[ j \left( \frac{2}{3} q^{3/2} - \frac{5\pi}{12} \right) \right] ; \\
 \frac{-2\pi}{3} &< \arg(q) < \frac{4\pi}{3} .
 \end{aligned}
 \tag{370}$$

To obtain the asymptotic expansion for  $k_2(q)$ , it is more convenient to express  $k_2(q)$  in terms of  $H_{1/3}^{(2)}(\cdot)$ . From Equations 356 and 358,

$$\begin{aligned}
 Ai(q) &= \frac{1}{\pi} \left(\frac{q}{3}\right)^{1/2} \left(-j\frac{\pi}{2}\right) e^{-j\pi/6} H_{1/3}^{(2)} \left( \frac{2}{3} q^{3/2} e^{-j\pi/2} \right) \\
 &= \frac{-j}{2} \left(\frac{q}{3}\right)^{1/2} e^{-j\pi/6} H_{1/3}^{(2)} \left( \frac{2}{3} q^{3/2} e^{-j\pi/2} \right) .
 \end{aligned}
 \tag{371}$$

Thus,

$$\begin{aligned}
 Ai(-q) &= -\frac{j}{2} \left(\frac{q}{3}\right)^{1/2} e^{-j\pi/2} e^{-j\pi/6} H_{1/3}^{(2)} \left( \frac{2}{3} q^{3/2} e^{-j3\pi/2} e^{-j\pi/2} \right) \\
 &= -\frac{1}{2} \left(\frac{q}{3}\right)^{1/2} e^{-j\pi/6} H_{1/3}^{(2)} \left( \frac{2}{3} q^{3/2} e^{-j2\pi} \right) .
 \end{aligned}
 \tag{372}$$

Substitute Equation 372 into 365 gives

$$k_2(q) = - \left( \frac{2}{3} q^{3/2} \right)^{1/2} H_{1/3}^{(2)} \left( \frac{2}{3} q^{3/2} e^{-j2\pi} \right) .
 \tag{373}$$

From Equation 369 and 373, the asymptotic expansion for  $k_2(q)$  is



$$\begin{aligned}
k_2(q) &\sim - \left( \frac{2}{3} q^{3/2} \right)^{1/3} \left( \frac{2}{\pi} \right)^{1/2} \left( \frac{2}{3} q^{3/2} e^{-j2\pi} \right)^{-1/2} \\
&\exp \left[ -j \left( \frac{2}{3} q^{3/2} e^{-j2\pi} - \frac{5\pi}{12} \right) \right] \\
&\sim 12^{1/6} \pi^{-1/2} q^{-1/4} \exp \left[ -j \left( \frac{2}{3} q^{3/2} - \frac{5\pi}{12} \right) \right] : \\
0 &< \arg(q) < 2\pi .
\end{aligned}
\tag{374}$$

## APPENDIX B. SAMPLE PRINTOUT OF INPUT AND OUTPUT FILES

\*\*\*\*\*Sample printout for a 2 m evaporation duct input file\*\*\*\*\*

```

9ghz02m
0
9600.00    FQMZIN
0.000000   DELFQ
1.000000   NFREQ
0.000000   MPOL
5.000000   ALOSS
15.000000  SEATMP
35.000000  SEASLT
1.000000   IFLAGE
0.000000   AIRTMP
0.000000   RH
0.000000   WGPM3
0.250000   RMSBHT
48.000000  ZTINIT
0.000000   DELZT
1.000000   NZT
46.000000  ZRINIT
7.000000   DELZR
2.000000   NZR
18.5000    XINIT
9.250000   DELX
3.000000   NY
0.000000   ZREF
16          NZLAYR
0.000000   zi[ 0]
0.000000   zim[ 0]
0.000000   zigab[ 0]
0.020000   zi[ 1]
-0.400000  zim[ 1]
0.000000   zigab[ 1]
0.040000   zi[ 2]
-0.550000  zim[ 2]
0.000000   zigab[ 2]
0.079000   zi[ 3]
-0.700000  zim[ 3]
0.000000   zigab[ 3]
0.158000   zi[ 4]
-0.850000  zim[ 4]
0.000000   zigab[ 4]
0.251000   zi[ 5]
-0.940000  zim[ 5]
0.000000   zigab[ 5]
0.631000   zi[ 6]
-1.100000  zim[ 6]
0.000000   zigab[ 6]
1.259000   zi[ 7]

```

-1.170000	zim[ 7]
0.000000	zigab[ 7]
2.000000	zi[ 8]
-1.180000	zim[ 8]
0.000000	zigab[ 8]
3.981000	zi[ 9]
-1.090000	zim[ 9]
0.000000	zigab[ 9]
5.012000	zi[ 10]
-1.010000	zim[ 10]
0.000000	zigab[ 10]
7.943000	zi[ 11]
-0.750000	zim[ 11]
0.000000	zigab[ 11]
12.589000	zi[ 12]
-0.270000	zim[ 12]
0.000000	zigab[ 12]
19.953000	zi[ 13]
0.550000	zim[ 13]
0.000000	zigab[ 13]
25.119000	zi[ 14]
1.140000	zim[ 14]
0.000000	zigab[ 14]
31.623000	zi[ 15]
1.900000	zim[ 15]
0.000000	zigab[ 15]
50.000000	zi[ 16]
4.069000	zim[ 16]
0.000000	zigab[ 16]

\*\*\*\*\*Sample printout for 9ghz02.eig\*\*\*\*\*

9ghz02m

1	
9600.000	FQMZIN
0.000000	DELFQ
1	NFREQ
0	MPOL
5.000000	ALOSS
15.000000	SEATMP
35.000000	SEASLT
1	IFLAGB
0.000000	AIRTMP
0.000000	RH
0.000000	WGPM3
0.250000	RMSBHT
48.000000	ZTINIT
0.000000	DELZT
1	NZT
46.000000	ZRINIT
7.000000	DELZR
2	NZR
18.500000	XINIT
9.250000	DELX
3	NX

```

0.000000  ZREF
16        NZLAYR
0.000000  zi[ 0]
0.000000  zim[ 0]
0.000000  zigab[ 0]
0.020000  zi[ 1]
-0.400000 zim[ 1]
0.000000  zigab[ 1]
0.040000  zi[ 2]
-0.550000 zim[ 2]
0.000000  zigab[ 2]
0.079000  zi[ 3]
-0.700000 zim[ 3]
0.000000  zigab[ 3]
0.158000  zi[ 4]
-0.850000 zim[ 4]
0.000000  zigab[ 4]
0.251000  zi[ 5]
-0.940000 zim[ 5]
0.000000  zigab[ 5]
0.631000  zi[ 6]
-1.100000 zim[ 6]
0.000000  zigab[ 6]
1.259000  zi[ 7]
-1.170000 zim[ 7]
0.000000  zigab[ 7]
2.000000  zi[ 8]
-1.180000 zim[ 8]
0.000000  zigab[ 8]
3.981000  zi[ 9]
-1.090000 zim[ 9]
0.000000  zigab[ 9]
5.012000  zi[ 10]
-1.010000 zim[ 10]
0.000000  zigab[ 10]
7.943000  zi[ 11]
-0.750000 zim[ 11]
0.000000  zigab[ 11]
12.589000 zi[ 12]
-0.270000 zim[ 12]
0.000000  zigab[ 12]
19.953000 zi[ 13]
0.550000  zim[ 13]
0.000000  zigab[ 13]
25.119000 zi[ 14]
1.140000  zim[ 14]
0.000000  zigab[ 14]
31.623000 zi[ 15]
1.900000  zim[ 15]
0.000000  zigab[ 15]
50.000000 zi[ 16]
4.069000  zim[ 16]
0.000000  zigab[ 16]
9         nrmode
(-1.269556983588969E-001,1.637613838717031E-001)  [ 1]
(-9.291194300692691E-002,1.251243717867209E-001)  [ 2]

```

```
(-4.143216670274859E-002,8.947793427709196E-002) [3]
(8.240706115401963E-003,6.917378956118456E-002) [4]
(5.551160395245244E-002,5.324818113693566E-002) [5]
(7.682611348504449E-002,6.708823745503940E-002) [6]
(1.394420773218969E-001,9.725689299213881E-002) [7]
(2.102775842495503E-001,1.357828223763695E-001) [8]
(2.864589183106626E-001,1.658090146540282E-001) [9]
```

\*\*\*\*\*Sample printout for 9ghz02m.out\*\*\*\*\*

1

```
frequency = 9600.0000 mhz
horizontal polarization
maximum mode attenuation rate = 5.0000 db/km

seawater temperature = 15.0000 degrees celsius
seawater salinity = 35.0000 grams salt/kg seawater
dielectric constant of seawater = 80.8869
conductivity of seawater = 4.6400 si/m
rms surface bump height = .2500 meters
iflagb = 1
air temperature = .0000 degrees celsius
relative humidity = .0000 percent
liquid water concentration in air = .0000 grams/meter***3

m( .0000 ) = .0000 m-units
gab( .0000 ) = .0000 db/km
```

#### tropospheric modified refractivity profile

z (meters)	m(z) (m-units)	gab(z) (db/km)	dm/dz (m-units/meter)	d(gab)/dz ((db/km)/meter)
.0000	.0000	.00000D+00	-20.0000	.00000D+00
.0200	-.4000	.00000D+00	-7.5000	.00000D+00
.0400	-.5500	.00000D+00	-3.8462	.00000D+00
.0790	-.7000	.00000D+00	-1.8987	.00000D+00
.1580	-.8500	.00000D+00	-.9677	.00000D+00
.2510	-.9400	.00000D+00	-.4211	.00000D+00
.6310	-1.1000	.00000D+00	-.1115	.00000D+00
1.2590	-1.1700	.00000D+00	-.0135	.00000D+00
2.0000	-1.1800	.00000D+00	.0454	.00000D+00
3.9810	-1.0900	.00000D+00	.0776	.00000D+00
5.0120	-1.0100	.00000D+00	.0887	.00000D+00
7.9430	-.7500	.00000D+00	.1033	.00000D+00
12.5890	-.2700	.00000D+00	.1114	.00000D+00
19.9530	.5500	.00000D+00	.1142	.00000D+00
25.1190	1.1400	.00000D+00	.1169	.00000D+00
31.6230	1.9000	.00000D+00	.1180	.00000D+00
50.0000	4.0690	.00000L+00		

\*\*\*\*\* start search for modal eigenvalues \*\*\*\*\*

tmesh= 1.67984D-03 tol= 1.00000D-04

zeros found in expanded search rectangle defined by  
tleft= 0.00D+00 tright= 2.69D-01  
ttop = 1.68D-01 tbot = -1.00D-04

qeigen( 1) = 8.24071D-03 6.91738D-02  
qeigen( 2) = 5.55116D-02 5.32482D-02  
qeigen( 3) = 2.10278D-01 1.35783D-01  
qeigen( 4) = 1.39442D-01 9.72569D-02  
qeigen( 5) = 7.68261D-02 6.70882D-02

zeros found in expanded search rectangle defined by  
tleft= -2.69D-01 tright= 0.00D+00  
ttop = 1.68D-01 tbot = -1.00D-04

qeigen( 1) = -1.26956D-01 1.63761D-01  
qeigen( 2) = -9.29119D-02 1.25124D-01  
qeigen( 3) = -4.14322D-02 8.94779D-02

zeros found in expanded search rectangle defined by  
tleft= -5.38D-01 tright= -2.69D-01  
ttop = 1.68D-01 tbot = -1.00D-04

zeros found in expanded search rectangle defined by  
tleft= -8.06D-01 tright= -5.38D-01  
ttop = 1.68D-01 tbot = -1.00D-04

zeros found in expanded search rectangle defined by  
tleft= 2.69D-01 tright= 5.38D-01  
ttop = 1.68D-01 tbot = -1.00D-04

qeigen( 1) = 2.86459D-01 1.65809D-01

zeros found in expanded search rectangle defined by  
tleft= 5.38D-01 tright= 8.06D-01  
ttop = 1.68D-01 tbot = -1.00D-04

```

zeros round in expanded search rectangle defined by
tleft=      8.06D-01      tright=      1.08D+00
ttop =      1.68D-01      tbot  =      -1.00D-04

```

```

***** modal eigenvalues  nrmode=  9 *****

```

mode	eigenvalue	theta	atnu db/km
1	-1.26956D-01 1.63761D-01	1.16912D-03 2.38566D-03	4.8743D+00
2	-9.29119D-02 1.25124D-01	1.03533D-03 2.05835D-03	3.7243D+00
3	-4.14322D-02 8.94779D-02	9.86784D-04 1.54436D-03	2.6633D+00
4	8.24071D-03 6.91738D-02	1.15188D-03 1.02280D-03	2.0589D+00
5	5.55116D-02 5.32482D-02	1.50185D-03 6.03859D-04	1.5849D+00
6	7.68261D-02 6.70882D-02	1.74517D-03 6.54734D-04	1.9969D+00
7	1.39442D-01 9.72569D-02	2.29575D-03 7.21529D-04	2.8948D+00
8	2.10278D-01 1.35783D-01	2.80081D-03 8.25696D-04	4.0415D+00
9	2.86459D-01 1.65809D-01	3.24286D-03 8.70843D-04	4.9353D+00

```

*****Sample printout for 9ghz02m.plt*****

```

3	1	2	frequency =	9600.0000 mhz				
18.5	48.0	46.0	52.90	61.21	84.54	76.23	56.5	
18.5	48.0	53.0	69.58	73.08	67.86	64.36	58.6	
27.8	48.0	46.0	2.17	38.48	138.80	102.48	56.5	
27.8	48.0	53.0	19.57	43.39	121.39	97.57	58.6	
37.0	48.0	46.0	3.41	22.15	140.05	121.31	56.5	
37.0	48.0	53.0	2.69	25.91	140.77	117.55	58.6	

```

*****Sample printout for a 14 m evaporation duct input file*****

```

```

9ghz14m
0
9600.00    FQMZIN
0.000000   DELFQ
1          NFREQ
0          MPOL
2.100000   ALOSS
15.000000  SEATMP
35.000000  SEASLT
1          IFLAGB
0.000000   AIRTMP
0.000000   RH
0.000000   WGPM3
0.250000   RMSBHT
25.000000  ZTINIT
0.000000   DELZT
1          NZT

```

3.000000	ZRINIT
7.000000	DELZR
2	NZR
18.5000	XINIT
9.250000	DELX
3	NX
0.000000	ZREF
20	NZLAYR
0.000000	zi[ 0]
0.000000	zim[ 0]
0.000000	zigab[ 0]
0.040000	zi[ 1]
-3.900000	zim[ 1]
0.000000	zigab[ 1]
0.100000	zi[ 2]
-5.340000	zim[ 2]
0.000000	zigab[ 2]
0.200000	zi[ 3]
-6.400000	zim[ 3]
0.000000	zigab[ 3]
0.398000	zi[ 4]
-7.460000	zim[ 4]
0.000000	zigab[ 4]
0.794000	zi[ 5]
-8.490000	zim[ 5]
0.000000	zigab[ 5]
1.585000	zi[ 6]
-9.470000	zim[ 6]
0.000000	zigab[ 6]
3.162000	zi[ 7]
-10.350000	zim[ 7]
0.000000	zigab[ 7]
6.310000	zi[ 8]
-11.040000	zim[ 8]
0.000000	zigab[ 8]
12.589000	zi[ 9]
-11.320000	zim[ 9]
0.000000	zigab[ 9]
14.000000	zi[ 10]
-11.330000	zim[ 10]
0.000000	zigab[ 10]
25.119000	zi[ 11]
-10.870000	zim[ 11]
0.000000	zigab[ 11]
39.811000	zi[ 12]
-9.750000	zim[ 12]
0.000000	zigab[ 12]
50.119000	zi[ 13]
-8.820000	zim[ 13]
0.000000	zigab[ 13]
63.096000	zi[ 14]
-7.560000	zim[ 14]
0.000000	zigab[ 14]
79.433000	zi[ 15]
-5.880000	zim[ 15]
0.000000	zigab[ 15]



```

100.000000 zi[ 16]
-3.670000 zim[ 16]
0.000000 zigab[ 16]
125.699000 zi[ 17]
-0.800000 zim[ 17]
0.000000 zigab[ 17]
158.489000 zi[ 18]
2.920000 zim[ 18]
0.000000 zigab[ 18]
199.526000 zi[ 19]
7.690000 zim[ 19]
0.000000 zigab[ 19]
209.526000 zi[ 20]
8.870000 zim[ 20]
0.000000 zigab[ 20]

```

\*\*\*\*\*Sample printout for 9ghz14m.eig\*\*\*\*\*

9ghz14m

```

1
9600.000 FQMZIN
0.000000 DELFQ
1 NFREQ
0 MPOL
2.100000 ALOSS
15.000000 SEATMP
35.000000 SEASLT
1 IFLAGB
0.000000 AIRTMP
0.000000 RH
0.000000 WGPM3
0.250000 RMSBHT
25.000000 ZTINIT
0.000000 DELZT
1 NZT
3.000000 ZRINIT
7.000000 DELZR
2 NZR
18.500000 XINIT
9.250000 DELX
3 NX
0.000000 ZREF
20 NZLAYR
0.000000 zi[ 0]
0.000000 zim[ 0]
0.000000 zigab[ 0]
0.040000 zi[ 1]
-3.900000 zim[ 1]
0.000000 zigab[ 1]
0.100000 zi[ 2]
-5.340000 zim[ 2]
0.000000 zigab[ 2]
0.200000 zi[ 3]
-6.400000 zim[ 3]
0.000000 zigab[ 3]

```

0.398000	zi[ 4]	
-7.460000	zim[ 4]	
0.000000	zigab[ 4]	
0.724000	zi[ 5]	
-8.490000	zim[ 5]	
0.000000	zigab[ 5]	
1.585000	zi[ 6]	
-9.470000	zim[ 6]	
0.000000	zigab[ 6]	
3.162000	zi[ 7]	
-10.350000	zim[ 7]	
0.000000	zigab[ 7]	
6.310000	zi[ 8]	
-11.040000	zim[ 8]	
0.000000	zigab[ 8]	
12.589000	zi[ 9]	
-11.320000	zim[ 9]	
0.000000	zigab[ 9]	
14.000000	zi[ 10]	
-11.330000	zim[ 10]	
0.000000	zigab[ 10]	
25.119000	zi[ 11]	
-10.870000	zim[ 11]	
0.000000	zigab[ 11]	
39.811000	zi[ 12]	
-9.750000	zim[ 12]	
0.000000	zigab[ 12]	
50.119000	zi[ 13]	
-8.820000	zim[ 13]	
0.000000	zigab[ 13]	
63.096000	zi[ 14]	
-7.560000	zim[ 14]	
0.000000	zigab[ 14]	
79.433000	zi[ 15]	
-5.880000	zim[ 15]	
0.000000	zigab[ 15]	
100.000000	zi[ 16]	
-3.670000	zim[ 16]	
0.000000	zigab[ 16]	
125.893000	zi[ 17]	
-0.800000	zim[ 17]	
0.000000	zigab[ 17]	
158.489000	zi[ 18]	
2.920000	zim[ 18]	
0.000000	zigab[ 18]	
199.526000	zi[ 19]	
7.690000	zim[ 19]	
0.000000	zigab[ 19]	
209.525000	zi[ 20]	
8.870000	zim[ 20]	
0.000000	zigab[ 20]	
94	nrmode	
(-1.141392174382303E-001,2.381981532549529E-002)		[ 1]
(-9.791475196101274E-002,2.146753762359050E-002)		[ 2]
(-8.353759867176463E-002,1.997153241435371E-002)		[ 3]
(-7.039783167095078E-002,1.889606906408722E-002)		[ 4]

(-5.814228837496516E-002,1.806765450625147E-002) [5]  
 (-4.656628790733621E-002,1.728118397954079E-002) [6]  
 (-3.585330645452212E-002,1.663363181205510E-002) [7]  
 (-2.52774200023139'E-002,1.628348275301875E-002) [8]  
 (-1.546901363155011E-002,1.571655186158130E-002) [9]  
 (-5.541956071147925E-003,1.540239335815212E-002) [10]  
 (3.551321843705835E-003,1.518851207972415E-002) [11]  
 (1.275397200580210E-002,1.460556299546134E-002) [12]  
 (2.182707485088627E-002,1.455854335720260E-002) [13]  
 (3.011306010124748E-002,1.427613140712134E-002) [14]  
 (3.842803342065884E-002,1.365892306660126E-002) [15]  
 (4.694372655547260E-002,1.365320803509313E-002) [16]  
 (5.485441606530146E-002,1.365090923924493E-002) [17]  
 (6.227012336327741E-002,1.336276100865187E-002) [18]  
 (7.029778501629384E-002,1.282089227646639E-002) [19]  
 (7.803056175576653E-002,1.285572220605075E-002) [20]  
 (8.555171301174155E-002,1.287860792288040E-002) [21]  
 (9.226312048960586E-002,1.278978888881959E-002) [22]  
 (9.940292680300993E-002,1.210154187763285E-002) [23]  
 (1.068485002790371E-001,1.208486102139816E-002) [24]  
 (1.138252208858628E-001,1.195444121483925E-002) [25]  
 (1.207736300828756E-001,1.220952877564303E-002) [26]  
 (1.270094738812559E-001,1.222384497715726E-002) [27]  
 (1.334153343077380E-001,1.139138734459426E-002) [28]  
 (1.403978164901863E-001,1.116571468633296E-002) [29]  
 (1.467611452238766E-001,1.129961336802001E-002) [30]  
 (1.534013614885507E-001,1.116543866415548E-002) [31]  
 (1.596149004805981E-001,1.136779885138012E-002) [32]  
 (1.652780520579286E-001,1.153156478829780E-002) [33]  
 (1.709514646617846E-001,1.066116492781759E-002) [34]  
 (1.772646284812436E-001,1.023671860551962E-002) [35]  
 (1.836482895835790E-001,1.049763117145163E-002) [36]  
 (1.891728805710848E-001,1.025773869961451E-002) [37]  
 (1.951501729145819E-001,1.007563477610222E-002) [38]  
 (2.006781595657978E-001,1.041321459422345E-002) [39]  
 (2.059566903740310E-001,1.043172189566190E-002) [40]  
 (2.106263141043572E-001,9.926428640075490E-003) [41]  
 (2.157518751516274E-001,8.798215719693539E-003) [42]  
 (2.213117492755677E-001,9.015024671611194E-003) [43]  
 (2.273294238073336E-001,8.321071087819404E-003) [44]  
 (2.303987147631242E-001,1.185554142465963E-003) [45]  
 (2.318960198770044E-001,8.464799970637166E-003) [46]  
 (2.348881611845628E-001,7.882321847941567E-003) [47]  
 (2.402844396949530E-001,8.719487457459410E-003) [48]  
 (2.455147471247609E-001,9.692237120376304E-003) [49]  
 (2.499562362348160E-001,1.010888147000029E-002) [50]  
 (2.545987673232816E-001,1.010149239614689E-002) [51]  
 (2.600542439477720E-001,1.049907356151861E-002) [52]  
 (2.662691013974199E-001,1.044438363485770E-002) [53]  
 (2.726418617644147E-001,1.092098859083637E-002) [54]  
 (2.787900254603262E-001,1.175196099261803E-002) [55]  
 (2.849476164544477E-001,1.227969492073188E-002) [56]  
 (2.913747922726816E-001,1.246149094591169E-002) [57]  
 (2.980151522337324E-001,1.329730594318235E-002) [58]  
 (3.042875049575047E-001,1.319657577519411E-002) [59]  
 (3.113767357254904E-001,1.314377702678184E-002) [60]

(3.187819780335167E-001,1.333680648010814E-002)	[ 61]
(3.264578839391167E-001,1.384739430048439E-002)	[ 62]
(3.338112497216490E-001,1.464186747417008E-002)	[ 63]
(3.410941286084828E-001,1.490152638527356E-002)	[ 64]
(3.487677270086091E-001,1.524482539938137E-002)	[ 65]
(3.563811238953353E-001,1.551353021082856E-002)	[ 66]
(3.643646265942290E-001,1.538942161737147E-002)	[ 67]
(3.729775181063331E-001,1.559989625580370E-002)	[ 68]
(3.814931085621074E-001,1.618605966683290E-002)	[ 69]
(3.898891425944480E-001,1.689462812245438E-002)	[ 70]
(3.980319033943261E-001,1.714047724320524E-002)	[ 71]
(4.067283853697590E-001,1.729643131530817E-002)	[ 72]
(4.154949433609382E-001,1.751352926449171E-002)	[ 73]
(4.246400421526860E-001,1.750551050950182E-002)	[ 74]
(4.341831113214738E-001,1.789042401852184E-002)	[ 75]
(4.435147531419498E-001,1.866150403101799E-002)	[ 76]
(4.525289981946153E-001,1.909787541066863E-002)	[ 77]
(4.618630098407780E-001,1.919260738082309E-002)	[ 78]
(4.716181770152394E-001,1.927318684881558E-002)	[ 79]
(4.815020432944205E-001,1.952391397029549E-002)	[ 80]
(4.917304699196995E-001,1.972766534431011E-002)	[ 81]
(5.020225407049786E-001,2.044077417123271E-002)	[ 82]
(5.118252276793764E-001,2.102681780869346E-002)	[ 83]
(5.218948685680153E-001,2.102179462275660E-002)	[ 84]
(5.324815199710420E-001,2.111602176832850E-002)	[ 85]
(5.432604196740195E-001,2.136847129075478E-002)	[ 86]
(5.541811165125796E-001,2.172276354461229E-002)	[ 87]
(5.651832495410506E-001,2.227384315438282E-002)	[ 88]
(5.758688247777175E-001,2.293313938401881E-002)	[ 89]
(5.866533408029618E-001,2.286336319718634E-002)	[ 90]
(5.981376451733623E-001,2.292879795570507E-002)	[ 91]
(6.097267273995461E-001,2.326416356305734E-002)	[ 92]
(6.213712082142159E-001,2.370559660462316E-002)	[ 93]
(6.330414592077247E-001,2.426761624643026E-002)	[ 94]

\*\*\*\*\*Sample printout for 9ghz14m.out\*\*\*\*\*

1

```

frequency =      9600.0000 mhz
horizontal polarization
maximum mode attenuation rate =      2.1000 db/km

seawater temperature =      15.0000 degrees celsius
seawater salinity =      35.0000 grams salt/kg seawater
dielectric constant of seawater =      80.8869
conductivity of seawater =      4.6400 si/m
rms surface bump height =      .2500 meters
iflagb =      1
air temperature =      .0000 degrees celsius
relative humidity =      .0000 percent
liquid water concentration in air =      .0000 grams/meter**3

m(      .0000 ) =      .0000 m-units
gab(      .0000 ) =      .0000 db/km

```

tropospheric modified refractivity profile

z (meters)	m(z) (m-units)	gab(z) (db/km)	dm/dz (m-units/meter)	d(gab)/dz ((db/km)/meter)
.0000	.0000	.00000D+00	-97.5000	.00000D+00
.0400	-3.9000	.00000D+00	-24.0000	.00000D+00
.1000	-5.3400	.00000D+00	-10.6000	.00000D+00
.2000	-6.4000	.00000D+00	-5.3535	.00000D+00
.3980	-7.4600	.00000D+00	-2.6010	.00000D+00
.7940	-8.4900	.00000D+00	-1.2389	.00000D+00
1.5850	-9.4700	.00000D+00	-.5580	.00000D+00
3.1620	-10.3500	.00000D+00	-.2192	.00000D+00
6.3100	-11.0400	.00000D+00	-.0446	.00000D+00
12.5890	-11.3200	.00000D+00	-.0071	.00000D+00
14.0000	-11.3300	.00000D+00	.0414	.00000D+00
25.1190	-10.8700	.00000D+00	.0762	.00000D+00
39.8110	-9.7500	.00000D+00	.0902	.00000D+00
50.1190	-8.8200	.00000D+00	.0971	.00000D+00
63.0960	-7.5600	.00000D+00	.1028	.00000D+00
79.4330	-5.8800	.00000D+00	.1075	.00000D+00
100.0000	-3.6700	.00000D+00	.1108	.00000D+00
125.8930	-.8000	.00000D+00	.1141	.00000D+00
158.4890	2.9200	.00000D+00	.1162	.00000D+00
199.5260	7.6900	.00000D+00	.1180	.00000D+00
209.5260	8.8700	.00000D+00		

\*\*\*\*\* start search for modal eigenvalues \*\*\*\*\*

tmesh= 5.84278D-04 tol= 5.84278D-05

zeros found in expanded search rectangle defined by

tleft= 1.87D-01 tright= 2.80D-01  
ttop = 2.45D-02 tbot = -5.84D-05

qeigen( 1) = 1.89173D-01 1.02577D-02  
qeigen( 2) = 2.10626D-01 9.92643D-03  
qeigen( 3) = 2.30399D-01 1.18555D-03  
qeigen( 4) = 2.34888D-01 7.88232D-03  
qeigen( 5) = 2.78790D-01 1.17520D-02  
qeigen( 6) = 2.72642D-01 1.09210D-02  
qeigen( 7) = 2.66269D-01 1.04444D-02  
qeigen( 8) = 2.60054D-01 1.04991D-02  
qeigen( 9) = 2.54599D-01 1.01015D-02  
qeigen( 10) = 2.49956D-01 1.01089D-02  
qeigen( 11) = 2.45515D-01 9.69224D-03  
qeigen( 12) = 2.40284D-01 8.71949D-03  
qeigen( 13) = 2.31896D-01 8.46480D-03

```

      qeigen( 14) =    2.27329D-01    8.32107D-03
      qeigen( 15) =    2.21312D-01    9.01502D-03
      qeigen( 16) =    2.15752D-01    8.79822D-03
      qeigen( 17) =    2.05957D-01    1.04317D-02
      qeigen( 18) =    2.00678D-01    1.04132D-02
      qeigen( 19) =    1.95150D-01    1.00756D-02

```

```

zeros found in expanded search rectangle defined by
tleft=    9.35D-02    tright=    1.87D-01
ttop =    2.45D-02    tbot  =   -5.84D-05

```

```

      qeigen(  1) =    9.94029D-02    1.21015D-02
      qeigen(  2) =    1.13825D-01    1.19544D-02
      qeigen(  3) =    1.70951D-01    1.06612D-02
      qeigen(  4) =    1.83648D-01    1.04976D-02
      qeigen(  5) =    1.77265D-01    1.02367D-02
      qeigen(  6) =    1.65278D-01    1.15316D-02
      qeigen(  7) =    1.59615D-01    1.13678D-02
      qeigen(  8) =    1.53401D-01    1.11654D-02
      qeigen(  9) =    1.46761D-01    1.12996D-02
      qeigen( 10) =    1.40398D-01    1.11657D-02
      qeigen( 11) =    1.33415D-01    1.13914D-02
      qeigen( 12) =    1.27009D-01    1.22238D-02
      qeigen( 13) =    1.20774D-01    1.22095D-02
      qeigen( 14) =    1.06849D-01    1.20849D-02

```

```

zeros found in expanded search rectangle defined by
tleft=    0.00D+00    tright=    9.35D-02
ttop =    2.45D-02    tbot  =   -5.84D-05

```

```

      qeigen(  1) =    3.55132D-03    1.51885D-02
      qeigen(  2) =    3.84280D-02    1.36589D-02
      qeigen(  3) =    9.22631D-02    1.27898D-02
      qeigen(  4) =    8.55517D-02    1.28786D-02
      qeigen(  5) =    7.80306D-02    1.28557D-02
      qeigen(  6) =    7.02978D-02    1.28209D-02
      qeigen(  7) =    6.22701D-02    1.33628D-02
      qeigen(  8) =    5.48544D-02    1.36509D-02
      qeigen(  9) =    4.69437D-02    1.36532D-02
      qeigen( 10) =    3.01131D-02    1.42761D-02
      qeigen( 11) =    2.18271D-02    1.45585D-02
      qeigen( 12) =    1.27540D-02    1.46056D-02

```

```

zeros found in expanded search rectangle defined by
tleft=   -9.35D-02    tright=    0.00D+00
ttop =    2.45D-02    tbot  =   -5.84D-05

```

```

      qeigen(  1) =   -8.35376D-02    1.99715D-02
      qeigen(  2) =   -4.65663D-02    1.72812D-02
      qeigen(  3) =   -5.54196D-03    1.54024D-02

```

```

qeigen( 4) = -1.54690D-02  1.57166D-02
qeigen( 5) = -2.52774D-02  1.62835D-02
qeigen( 6) = -3.53533D-02  1.66336D-02
qeigen( 7) = -5.81423D-02  1.80677D-02
qeigen( 8) = -7.03978D-02  1.88961D-02

```

```

zeros found in expanded search rectangle defined by
tleft= -1.87D-01  tright= -9.35D-02
ttop =  2.45D-02  tbot = -5.84D-05

```

```

qeigen( 1) = -9.79148D-02  2.14675D-02
qeigen( 2) = -1.14139D-01  2.38198D-02

```

```

zeros found in expanded search rectangle defined by
tleft= -2.80D-01  tright= -1.87D-01
ttop =  2.45D-02  tbot = -5.84D-05

```

```

zeros found in expanded search rectangle defined by
tleft= -3.74D-01  tright= -2.80D-01
ttop =  2.45D-02  tbot = -5.84D-05

```

```

zeros found in expanded search rectangle defined by
tleft=  2.80D-01  tright=  3.74D-01
ttop =  2.45D-02  tbot = -5.84D-05

```

```

qeigen( 1) =  3.72978D-01  1.55999D-02
qeigen( 2) =  3.64365D-01  1.53894D-02
qeigen( 3) =  3.56381D-01  1.55135D-02
qeigen( 4) =  3.48768D-01  1.52448D-02
qeigen( 5) =  3.41094D-01  1.49015D-02
qeigen( 6) =  3.33811D-01  1.46419D-02
qeigen( 7) =  3.26458D-01  1.38474D-02
qeigen( 8) =  3.18782D-01  1.33368D-02
qeigen( 9) =  3.11377D-01  1.31438D-02
qeigen(10) =  3.04288D-01  1.31966D-02
qeigen(11) =  2.98015D-01  1.32973D-02
qeigen(12) =  2.91375D-01  1.24615D-02
qeigen(13) =  2.84948D-01  1.22797D-02

```

```

zeros found in expanded search rectangle defined by
tleft=  3.74D-01  tright=  4.67D-01
ttop =  2.45D-02  tbot = -5.84D-05

```

```

qeigen( 1) =  4.61863D-01  1.91926D-02

```

```

qeigen( 2) = 4.52529D-01 1.90979D-02
qeigen( 3) = 4.43515D-01 1.86615D-02
qeigen( 4) = 4.34183D-01 1.78904D-02
qeigen( 5) = 4.24640D-01 1.75055D-02
qeigen( 6) = 4.15495D-01 1.75135D-02
qeigen( 7) = 4.06728D-01 1.72964D-02
qeigen( 8) = 3.98032D-01 1.71405D-02
qeigen( 9) = 3.89889D-01 1.68946D-02
qeigen(10) = 3.81493D-01 1.61861D-02

```

```

zeros found in expanded search rectangle defined by
tleft= 4.67D-01 tright= 5.61D-01
ttop = 2.45D-02 tbot = -5.84D-05

```

```

qeigen( 1) = 5.54181D-01 2.17228D-02
qeigen( 2) = 5.43260D-01 2.13685D-02
qeigen( 3) = 5.32482D-01 2.11160D-02
qeigen( 4) = 5.21895D-01 2.10218D-02
qeigen( 5) = 5.11825D-01 2.10268D-02
qeigen( 6) = 5.02023D-01 2.04408D-02
qeigen( 7) = 4.91730D-01 1.97277D-02
qeigen( 8) = 4.81502D-01 1.95239D-02
qeigen( 9) = 4.71618D-01 1.92732D-02

```

```

zeros found in expanded search rectangle defined by
tleft= 5.61D-01 tright= 6.54D-01
ttop = 2.45D-02 tbot = -5.84D-05

```

```

qeigen( 1) = 6.44509D-01 2.47339D-02
qeigen( 2) = 6.33041D-01 2.42676D-02
qeigen( 3) = 6.21371D-01 2.37056D-02
qeigen( 4) = 6.09727D-01 2.32642D-02
qeigen( 5) = 5.98138D-01 2.29288D-02
qeigen( 6) = 5.86653D-01 2.28634D-02
qeigen( 7) = 5.75869D-01 2.29331D-02
qeigen( 8) = 5.65183D-01 2.22738D-02

```

```

zeros found in expanded search rectangle defined by
tleft= 6.54D-01 tright= 7.48D-01
ttop = 2.45D-02 tbot = -5.84D-05

```

2 modes found on same phase line

```

qeigen( 1) = 6.68467D-01 2.47500D-02
qeigen( 2) = 6.56153D-01 2.47347D-02

```

```

zeros found in expanded search rectangle defined by
tleft= 7.48D-01 tright= 8.41D-01

```



ttop = 2.45D-02 tbot = -5.84D-03

2 modes found on same phase line

\*\*\*\*\* modal eigenvalues nrmode= 94 \*\*\*\*\*

mode	eigenvalue	theta	atnu db/km
1	-1.14139D-01 2.38198D-02	3.47000D-04 3.36133D-03	2.0384D+00
2	-9.79148D-02 2.14675D-02	3.37466D-04 3.11498D-03	1.8371D+00
3	-8.35376D-02 1.99715D-02	3.39522D-04 2.88036D-03	1.7091D+00
4	-7.03978D-02 1.88961D-02	3.49316D-04 2.64885D-03	1.6170D+00
5	-5.81423D-02 1.80677D-02	3.66463D-04 2.41421D-03	1.5461D+00
6	-4.65663D-02 1.72812D-02	3.89814D-04 2.17081D-03	1.4788D+00
7	-3.58533D-02 1.66336D-02	4.23953D-04 1.92120D-03	1.4234D+00
8	-2.52774D-02 1.62835D-02	4.84349D-04 1.64625D-03	1.3935D+00
9	-1.54690D-02 1.57166D-02	5.67769D-04 1.35548D-03	1.3450D+00
10	-5.54196D-03 1.54024D-02	7.28131D-04 1.03582D-03	1.3181D+00
11	3.55132D-03 1.51885D-02	9.68348D-04 7.68051D-04	1.2998D+00
12	1.27540D-02 1.46056D-02	1.25460D-03 5.70059D-04	1.2499D+00
13	2.18271D-02 1.45585D-02	1.53413D-03 4.64688D-04	1.2459D+00
14	3.01131D-02 1.42761D-02	1.76251D-03 3.96631D-04	1.2217D+00
15	3.84280D-02 1.36589D-02	1.96946D-03 3.39607D-04	1.1689D+00
16	4.69437D-02 1.36532D-02	2.16626D-03 3.08626D-04	1.1684D+00
17	5.48544D-02 1.36509D-02	2.33540D-03 2.86226D-04	1.1682D+00
18	6.22701D-02 1.33628D-02	2.48351D-03 2.63474D-04	1.1435D+00
19	7.02978D-02 1.28209D-02	2.63465D-03 2.38289D-04	1.0972D+00
20	7.80306D-02 1.28557D-02	2.77370D-03 2.26958D-04	1.1001D+00
21	8.55517D-02 1.28786D-02	2.90271D-03 2.17257D-04	1.1021D+00
22	9.22631D-02 1.27898D-02	3.01313D-03 2.07852D-04	1.0945D+00
23	9.94029D-02 1.21015D-02	3.12585D-03 1.89575D-04	1.0356D+00
24	1.06849D-01 1.20849D-02	3.23999D-03 1.82645D-04	1.0342D+00
25	1.13825D-01 1.19544D-02	3.34337D-03 1.75087D-04	1.0230D+00
26	1.20774D-01 1.22095D-02	3.44356D-03 1.73620D-04	1.0448D+00
27	1.27009D-01 1.22238D-02	3.53092D-03 1.69523D-04	1.0461D+00
28	1.33415D-01 1.13914D-02	3.61798D-03 1.54177D-04	9.7483D-01
29	1.40398D-01 1.11657D-02	3.71101D-03 1.47335D-04	9.5552D-01
30	1.46761D-01 1.12996D-02	3.79399D-03 1.45841D-04	9.6698D-01
31	1.53401D-01 1.11654D-02	3.87856D-03 1.40966D-04	9.5550D-01
32	1.59615D-01 1.13678D-02	3.95622D-03 1.40704D-04	9.7282D-01
33	1.65278D-01 1.15316D-02	4.02569D-03 1.40268D-04	9.8683D-01
34	1.70951D-01 1.06612D-02	4.09370D-03 1.27526D-04	9.1235D-01
35	1.77265D-01 1.02367D-02	4.16832D-03 1.20257D-04	8.7602D-01
36	1.83648D-01 1.04976D-02	4.24268D-03 1.21161D-04	8.9835D-01
37	1.89173D-01 1.02577D-02	4.30584D-03 1.16655D-04	8.7782D-01
38	1.95150D-01 1.00756D-02	4.37319D-03 1.12820D-04	8.6224D-01
39	2.00678D-01 1.04132D-02	4.43471D-03 1.14982D-04	8.9113D-01
40	2.05957D-01 1.04317D-02	4.49259D-03 1.13703D-04	8.9271D-01
41	2.10626D-01 9.92643D-03	4.54304D-03 1.06994D-04	8.4947D-01
42	2.15752D-01 8.79822D-03	4.59766D-03 9.37065D-05	7.5292D-01
43	2.21312D-01 9.01502D-03	4.65652D-03 9.48020D-05	7.7148D-01
44	2.27329D-01 8.32107D-03	4.71922D-03 8.63419D-05	7.1209D-01

45	2.30399D-01	1.18555D-03	4.75019D-03	1.22215D-05	1.0146D-01
46	2.31896D-01	8.46480D-03	4.76638D-03	8.69643D-05	7.2439D-01
47	2.34888D-01	7.88232D-03	4.79691D-03	8.04647D-05	6.7454D-01
48	2.40284D-01	8.71949D-03	4.85181D-03	8.80035D-05	7.4619D-01
49	2.45515D-01	9.69224D-03	4.90448D-03	9.67707D-05	8.2943D-01
50	2.49956D-01	1.01089D-02	4.94869D-03	1.00029D-04	8.6509D-01
51	2.54599D-01	1.01015D-02	4.99440D-03	9.90411D-05	8.6445D-01
52	2.60054D-01	1.04991D-02	5.04766D-03	1.01853D-04	8.9848D-01
53	2.66269D-01	1.04444D-02	5.10756D-03	1.00134D-04	8.9380D-01
54	2.72642D-01	1.09210D-02	5.16837D-03	1.03472D-04	9.3458D-01
55	2.78790D-01	1.17520D-02	5.22643D-03	1.10108D-04	1.0057D+00
56	2.84948D-01	1.22797D-02	5.28389D-03	1.13802D-04	1.0509D+00
57	2.91375D-01	1.24615D-02	5.34313D-03	1.14206D-04	1.0664D+00
58	2.98015D-01	1.32973D-02	5.40378D-03	1.20498D-04	1.1379D+00
59	3.04288D-01	1.31966D-02	5.46027D-03	1.18348D-04	1.1293D+00
60	3.11377D-01	1.31438D-02	5.52345D-03	1.16527D-04	1.1248D+00
61	3.18782D-01	1.33368D-02	5.58872D-03	1.16857D-04	1.1413D+00
62	3.26458D-01	1.38474D-02	5.65564D-03	1.19895D-04	1.1850D+00
63	3.33811D-01	1.46419D-02	5.71907D-03	1.25368D-04	1.2530D+00
64	3.41094D-01	1.49015D-02	5.78111D-03	1.26222D-04	1.2752D+00
65	3.48768D-01	1.52448D-02	5.84578D-03	1.27702D-04	1.3046D+00
66	3.56381D-01	1.55135D-02	5.90923D-03	1.28557D-04	1.3276D+00
67	3.64365D-01	1.53894D-02	5.97497D-03	1.26126D-04	1.3170D+00
68	3.72978D-01	1.55999D-02	6.04515D-03	1.26366D-04	1.3350D+00
69	3.81493D-01	1.61861D-02	6.11381D-03	1.29642D-04	1.3852D+00
70	3.89889D-01	1.68946D-02	6.18078D-03	1.33851D-04	1.4458D+00
71	3.98032D-01	1.71405D-02	6.24497D-03	1.34403D-04	1.4668D+00
72	4.06728D-01	1.72964D-02	6.31279D-03	1.34169D-04	1.4802D+00
73	4.15495D-01	1.75135D-02	6.38044D-03	1.34413D-04	1.4988D+00
74	4.24640D-01	1.75055D-02	6.45021D-03	1.32898D-04	1.4981D+00
75	4.34183D-01	1.78904D-02	6.52228D-03	1.34320D-04	1.5310D+00
76	4.43515D-01	1.86615D-02	6.59206D-03	1.38626D-04	1.5970D+00
77	4.52529D-01	1.90979D-02	6.65873D-03	1.40447D-04	1.6344D+00
78	4.61863D-01	1.91926D-02	6.72700D-03	1.39711D-04	1.6425D+00
79	4.71618D-01	1.92732D-02	6.79763D-03	1.38840D-04	1.6494D+00
80	4.81502D-01	1.95239D-02	6.86847D-03	1.39196D-04	1.6708D+00
81	4.91730D-01	1.97277D-02	6.94101D-03	1.39179D-04	1.6883D+00
82	5.02023D-01	2.04408D-02	7.01331D-03	1.42723D-04	1.7493D+00
83	5.11825D-01	2.10268D-02	7.08148D-03	1.45402D-04	1.7994D+00
84	5.21895D-01	2.10218D-02	7.15075D-03	1.43959D-04	1.7990D+00
85	5.32482D-01	2.11160D-02	7.22286D-03	1.43161D-04	1.8071D+00
86	5.43260D-01	2.13685D-02	7.29558D-03	1.43428D-04	1.8287D+00
87	5.54181D-01	2.17228D-02	7.36854D-03	1.44363D-04	1.8590D+00
88	5.65183D-01	2.22738D-02	7.44134D-03	1.46577D-04	1.9062D+00
89	5.75869D-01	2.29331D-02	7.51138D-03	1.49509D-04	1.9626D+00
90	5.86653D-01	2.28634D-02	7.58133D-03	1.47679D-04	1.9566D+00
91	5.98138D-01	2.29288D-02	7.65513D-03	1.46674D-04	1.9622D+00
92	6.09727D-01	2.32642D-02	7.72892D-03	1.47398D-04	1.9909D+00
93	6.21371D-01	2.37056D-02	7.80238D-03	1.48781D-04	2.0287D+00
94	6.33041D-01	2.42676D-02	7.87532D-03	1.50898D-04	2.0768D+00

\*\*\*\*\*Sample printout for 9ghz14m.plt\*\*\*\*\*

3 1 2  
frequency = 9600.0000 mhz

18.5	25.0	3.0	3.71	-.11	133.73	137.55	27.7
18.5	25.0	10.0	-8.31	.94	145.74	136.49	33.6
27.8	25.0	3.0	1.66	-.54	139.30	141.50	27.7
27.8	25.0	10.0	-1.61	.84	142.57	140.12	33.6
37.0	25.0	3.0	.08	-.58	143.38	144.04	27.7
37.0	25.0	10.0	.85	.90	142.61	142.56	33.6

## LIST OF REFERENCES

1. G.B. Baumgartner Jr, "XWVG : A waveguide program for trilinear tropospheric ducts," Technical Document 610, Naval Ocean Systems Center, San Diego, CA 92152, June 1983.
2. R. Pappert, "Field strength and path loss in a multilayer tropospheric waveguide environment," Technical Note 1366, Naval Ocean Systems Center, San Diego, CA 92152, October 1984.
3. C.L. Pekeris, "Accuracy of the earth-flattening approximation in the theory of microwave propagation," *Physical Review* , Vol. 70, No. 7 and 8, pp. 518-522, October 1946.
4. Milton Abramowitz and Irene A. Stegun, *Handbook of Mathematical Functions* , Dover Publications, Inc., New York, NY 10014, December 1972.
5. Sherman Marcus and William D. Stuart, "A model to calculate EM fields in tropospheric duct environments at frequencies through SHF," Electromagnetic Compatibility Analysis Center Report ESD-TR-81-102, Annapolis, MD 21402, September 1981.
6. R. Pappert and C. Goodhart, "Waveguide calculations of signal levels in tropospheric ducting environments," Technical Note 3129, Naval Electronics Laboratory Center, San Diego, CA 92152, February 1976.
7. Franz E. Hohn, *Elementary Matrix Algebra* , 3rd Edition, Macmillan Company, New York, NY 10022, 1973.
8. P. Beckman and A. Spizzichino, *The Scattering of Electromagnetic Waves from Rough Surfaces* , Macmillan, New York, NY 10022, 1963.
9. C.H. Collie, J.B. Hasted and D.M. Ritson, "The dielectric properties of water and heavy water," *Proc Phys Soc London* , Vol. 60, pp. 145-160, 1948.
10. Lawrence A. Klein and Calvin T. Swift, "An improved model for the dielectric constant of sea water at microwave frequencies," *IEEE Trans. on Antennas and Propagation* , Vol. 25, No. 1, pp. 104-111, January 1977.
11. Hans J. Liebe, "Modelling attenuation and phase of radio waves in air at frequencies below 1000 GHz," *Radio Science* , Vol. 16, No. 6, pp. 1183-1199, November-December 1981.
12. Lamont V. Blake, *Radar Range-Performance Analysis* , Artech House Inc., Norwood, MA 02062, 1986.
13. Donald E. Kerr, *Propagation of Short Radio Waves* , Peter Peregrinus Ltd., London, United Kingdom, 1987.
14. D.G. Morfitt and C.H. Shellman, "MODESRCH, An improved computer program for obtaining ELF-VLF-LF mode constants in an earth-ionosphere waveguide," Interim Report 77T, Naval Ocean Systems Center, San Diego, CA 92152, October 1976.

15. Lester A. Rubinfeld, *A First Course in Applied Complex Variables*, John Wiley & Sons Inc., 1985.
16. Z. Schulten and D.G.M. Anderson, "An algorithm for the evaluation of the complex Airy functions," *Journal of Computational Physics*, Vol. 31, No. 60-75, pp. 60-75, 1979.
17. Herbert V. Hitney, Juergen H. Richter, Richard A. Pappert, Kenneth D. Anderson and Gerald B. Baumgartner, Jr., "Tropospheric radio propagation assessment," *Proceedings of the IEEE*, Vol. 73, No. 2, pp.265-283, February 1985.
18. K.D. Anderson, "Low-altitude millimeter-wave propagation in the evaporation duct," Technical Report 1300, Naval Ocean Systems Center, San Diego, CA 92152, July 1989.
19. S.L. Broschat, E.I. Thorsos, A. Ishimaru, "The phase perturbation techniques vs. an exact numerical method for random rough surface scattering," *Journal of Electromagnetic Waves and Applications*, Vol. 3, No. 3, pp. 237-256, 1989.
20. Darrell R. Jackson, Dale P. Winebrenner and Akira Ishimaru, "Comparison of perturbation theories for rough-surface scattering," *J. Acoust. Soc. Am.*, No. 83(3), pp. 961-969, March 1988.
21. V. Blanchetiere-Ciarletti, J. Lavergnat, M. Sylvain and A. Weill, "Experimental observation of horizontal refractivity gradients during periods of multipath propagation," *Radio Science*, Vol. 24, No. 6, pp. 705-724, November-December 1989.
22. Se Hyun Cho and James R. Wait, "Electromagnetic wave propagation in a laterally non-uniform troposphere," *Radio Science*, Volume 13, Number 2, pp. 253, March-April 1978.
23. H.V. Hitney and R.A. Paulus, "Integrated refractive effects prediction system (IREPS), interim user's manual," Technical Document 238, Naval Ocean Systems Center, San Diego, CA 92152, March 1979.
24. H.V. Hitney, A.E. Barrios and G.E. Lindem, "Enginner's refractive effects prediction system (EREPS), revision 1.00 user's manual," Technical Document 1342, Naval Ocean Systems Center, San Diego, CA 92152, July 1988.

## INITIAL DISTRIBUTION LIST

	No. Copies
1. Defense Technical Information Center Cameron Station Alexandria, VA 22304-6145	2
2. Library, Code 0142 Naval Postgraduate School Monterey, CA 93943-5002	2
3. Research Administration, Code 012 Naval Postgraduate School Monterey, CA 93943-5000	1
4. Chairman, Code EC Department of Electrical and Computer Engineering Naval Postgraduate School Monterey, CA 93943-5000	1
5. Professor Hung-Mou Lee, Code EC Lh Naval Postgraduate School Monterey, CA 93943-5000	10
6. Professor Richard W. Adler, Code EC Ab Naval Postgraduate School Monterey, CA 93943-5000	2
7. Herbert V. Hitney Code 543 Naval Ocean Systems Center 271, Catalina Blvd San Diego, CA 92152-5000	1
8. Director, Defence Materials Organisation LEO Building Paya Lebar Airport Airport Road Singapore 1953 Republic of Singapore	4
9. Kuo-Huei Yen No. 12, Constitution Rd Lin-ya Area Kaoshiung R.O.C., Taiwan	1

10. Lean-Weng Yeoh  
#04-161, Blk 123  
Serangoon North Ave 1  
Singapore 1955  
Republic of Singapore

4